



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

NÁVRH SQL DATABÁZE PRO ELEKTRONICKÝ  
OBCHOD

PROPOSAL OF SQL DATABASE FOR E-COMMERCE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Tomáš Kadleček

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jiří Kříž, Ph.D.

BRNO 2017

## Zadání bakalářské práce

Ústav: Ústav informatiky  
Student: **Tomáš Kadleček**  
Studijní program: Systémové inženýrství a informatika  
Studijní obor: Manažerská informatika  
Vedoucí práce: **Ing. Jiří Kříž, Ph.D.**  
Akademický rok: 2016/17

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

### Návrh SQL databáze pro elektronický obchod

#### Charakteristika problematiky úkolu:

Úvod  
Cíle práce, metody a postupy zpracování  
Teoretická východiska práce  
Analýza současného stavu  
Vlastní návrhy řešení  
Závěr  
Seznam použité literatury  
Přílohy

#### Cíle, kterých má být dosaženo:

Cílem práce je návrh databáze pro potřeby elektronického obchodu na základě analýzy jeho potřeb.

#### Základní literární prameny:

CEJPEK, Jiří. Informace, komunikace a myšlení: úvod do informační vědy. 2. přeprac. vyd. Praha: Karolinum, 2005, 233 s. ISBN 80-246-1037-X.

KOCH, Miloš a Bernard NEUWIRTH. Datové a funkční modelování. Brno: Cerm, 2010. ISBN: 978-8-214-4125- 5.

KROENKE, David, David J. AUER a Jakub GONER. Databáze. Brno: Computer Press, 2015. 496 s. ISBN 978-80-251-4352-0.

LACKO, Ľuboslav. Mistrovství v SQL Server 2012. Brno: Computer Press, 2013. 640 s. ISBN 978-8-251-3773-4.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2016/17

V Brně dne 28.2.2017

L. S.

---

doc. RNDr. Bedřich Půža, CSc.  
ředitel

---

doc. Ing. et Ing. Stanislav Škapa, Ph.D.  
děkan

## **Abstrakt**

Tato bakalářská práce se zabývá návrhem a vytvořením SQL databáze pro elektronický obchod. Obsahuje teoretickou část, analýzu současného stavu a také praktickou část. V teoretické části jsou potřebné informace k pochopení databází. Analýza současného stavu je věnována již fungujícímu e-shopu, který slouží jako inspirace a zdroj informací pro praktickou část. Praktická část se věnuje vlastnímu návrhu řešení a zhodnocení přínosů, které z tvorby databáze plynou.

## **Abstract**

This bachelor thesis deals about proposal and creating SQL database for e-commerce. Including theoretical part, analysis of present conditions and requirements and also practical part. In theoretical part we can find necessary informations to understand the databases. Analysis of present conditions and requirements is focused to already working e-commerce, which is used as source of informations in practical part. Practical part is about own solution and benefits evaluation of database.

## **Klíčová slova**

sql, databáze, data, relace, elektronický obchod

## **Key words**

sql, databases, data, relation, e-commerce

### **Bibliografická citace**

KADLEČEK, T. *Návrh SQL databáze pro e-shop*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2017. 76s. Vedoucí bakalářské práce Ing. Jiří Kříž, Ph.D.

### **Čestné prohlášení**

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 31. května 2017

.....

Tomáš Kadleček

## **Poděkování**

Rád bych tímto poděkoval vedoucímu mé bakalářské práce panu Ing. Jířímu Křížovi, Ph.D. za odbornou pomoc, cenné rady a vstřícné vedení při zpracování této bakalářské práce.

# OBSAH

|  |    |
|--|----|
| ÚVOD.....  | 11 |
| 1 CÍL A METODIKA PRÁCE.....  | 12 |
| 2 TEORETICKÁ VÝCHODISKA PRÁCE.....                                     | 13 |
| 2.1 Data, informace a znalosti.....                                    | 13 |
| 2.1.1 Data.....  | 13 |
| 2.1.2 Informace.....   | 14 |
| 2.1.3 Znalosti.....  | 14 |
| 2.2 Informační technologie (IT).....                                   | 15 |
| 2.2.1 Hardware .....   | 15 |
| 2.2.2 Software.....  | 15 |
| 2.3 Databáze.....  | 15 |
| 2.3.1 Základní pojmy využívané v databázích .....                      | 16 |
| 2.3.2 Typy databází podle struktury dat .....                          | 19 |
| 2.4 Databázové systémy.....  | 23 |
| 2.4.1 Systém řízení databáze (DBMS – Database management system) ..... | 23 |
| 2.4.2 Uživatel.....  | 25 |
| 2.4.3 Databázová aplikace .....  | 25 |
| 2.4.4 Vlastní databázi .....   | 25 |
| 2.5 Jazyk SQL .....  | 25 |
| 2.5.1 Historie jazyka SQL .....  | 26 |
| 2.6 Vývojové prostředí MS SQL Server 2012 .....                        | 26 |
| 3 ANALÝZA SOUČASNÉHO STAVU.....  | 27 |
| 3.1 Základní údaje o firmě .....                                       | 27 |
| 3.2 Hlavní činnost .....   | 28 |
| 3.3 Historie.....  | 28 |
| 3.4 Organizační struktura .....  | 28 |
| 3.5 Konkurence .....   | 29 |
| 3.6 Zákazníci.....   | 29 |
| 3.7 Hardware.....  | 29 |



|       |  |    |
|-------|--|----|
| 3.8   | Software .....                                   | 30 |
| 3.8.1 | Operační systém .....                            | 30 |
| 3.9   | Základní procesy stávajícího řešení .....        | 31 |
| 3.9.1 | Registrace zákazníka .....                       | 31 |
| 3.9.2 | Vytvoření objednávky .....                       | 34 |
| 3.9.3 | Proces aktualizace dat.....                      | 36 |
| 4     | NÁVRH VLASTNÍHO ŘEŠENÍ.....                      | 38 |
| 4.1   | Základní procesy v databázi.....                 | 38 |
| 4.1.1 | Registrace a přihlášení zákazníka.....           | 38 |
| 4.1.2 | Vytvoření objednávky .....                       | 40 |
| 4.1.3 | Návrh na změnu procesu aktualizace dat .....     | 42 |
| 4.2   | Požadavky na databázi .....                      | 44 |
| 4.2.1 | Díličí požadavky na databázi.....                | 44 |
| 4.3   | Konceptuální návrh databáze .....                | 46 |
| 4.3.1 | Definice základních entit.....                   | 47 |
| 4.3.2 | Vztahy mezi základními entitami .....            | 49 |
| 4.4   | Logický návrh databáze .....                     | 51 |
| 4.4.1 | Dekompozice vazeb M:N .....                      | 51 |
| 4.4.2 | Přehled všech vytvořených tabulek .....          | 53 |
| 4.5   | Fyzický návrh databázemi.....                    | 60 |
| 4.5.1 | Volba rozhraní pro naprogramování databáze ..... | 61 |
| 4.5.2 | Triggery .....                                   | 61 |
| 4.5.3 | Pohledy.....                                     | 62 |
| 4.5.4 | Procedury.....                                   | 64 |
| 4.5.5 | Další postup po vlastním návrhu .....            | 65 |
| 4.5.6 | Zhodnocení a přínosy vlastního řešení .....      | 67 |
|       | ZÁVĚR .....                                      | 68 |
|       | SEZNAM POUŽITÝCH ZDROJŮ .....                    | 69 |
|       | SEZNAM OBRÁZKŮ.....                              | 72 |
|       | SEZNAM TABULEK .....                             | 73 |

|  |    |
|--|----|
| SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ ..... | 75 |
| SEZNAM PŘÍLOH.....                       | 76 |

## ÚVOD

Vědomosti o databázových technologiích dnes získávají větší a větší význam. Je to tím, že s daty pracujeme dnes a denně a je potřeba je někde ukládat. Abychom data mohli uložit, potřebujeme nějaké úložiště. Uložením to ovšem nekončí, surová data pro nás nemají takovou hodnotu jako data upravená. To vše, a ještě víc nám umožňují databáze.

Databáze se používají téměř všude. Setkáváme se s nimi už při klasickém ukládání dokumentů, dále pak ve složitější podobě jako základ elektronického obchodu a jiných webových aplikací. Databáze spočívají v jádru aplikací na řízení podniků a podporu rozhodování. Pomocí databází mohou lidé levněji a efektivněji udržovat přehled o nějakých skutečnostech. K databázím má dnes možnost přístupu skoro každý. Říká se, že v současnosti existuje více jak 10 miliónů aktivních databází.

Existují desítky databázových systémů. Nejznámějšími jsou Microsoft SQL Server, MySQL, Oracle, Firebird, PostgreSQL nebo třeba Microsoft Access. Každý databázový systém by měl nabízet základní funkce jako zajištění integrity dat, udržení dat podle nadefinovaných struktur, primární SQL jazyk pro práci s databází, základní zabezpečení nebo třeba již uložené procedury, funkce a trigger.

Volba databázového systému je na každém z nás. Důležité je, aby databázový systém splňoval vše, co požadujeme.

..

# 1 CÍL A METODIKA PRÁCE

Hlavním cílem bakalářské práce je navrhnout databázi pro elektronický obchod v MS SQL Server 2014. Navržené řešení by mělo přispět k zefektivnění práce a investic, zjednodušení některých procesů, zvýšení konkurenceschopnosti a ziskovosti e-shopu. Pro dosažení tohoto cíle, je potřeba definovat dílčí cíle.

Dílčí cíle práce jsou:

- na základě poznatků zpracovat teoretická východiska pro pochopení, návrh a tvorbu databáze,
- provést analýzu vnitřního a vnějšího prostředí, dále pak identifikace slabin a nedostatků základních procesů,
- navrhnout řešení na základě analýzy, na kterém bude moct e-shop stavět,
- vytvořit základní verzi databáze spolu s procedurami a pohledy nezbytnými k práci s databází,
- zhodnotit věcný i ekonomický přínos databáze.

## 2 TEORETICKÁ VÝCHODISKA PRÁCE

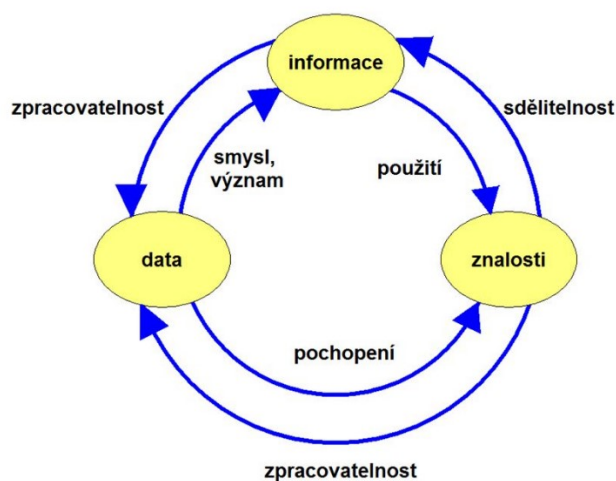
V této kapitole jsou popsány základní teoretické pojmy a metody spojené s databázemi a jejich tvorbou. Jsou důležité k porozumění problematiky a zvládnutí této práce.

### 2.1 Data, informace a znalosti

Uvedeme si, co to jsou data, informace a znalosti. Co nás na informacích nejvíce zajímá a jaké jsou typy znalostí.

#### 2.1.1 Data

Data nebo údaje jsou vjemy, které dokážeme zachytit svými smysly. Mají určitou vypo-  
vídající schopnost, mohou být nějak uspořádány, seřazeny a jsou uživateli dostupné  
v různých formách, jako jsou tabulky, grafy, zvukové signály a další (6).



Obr. 1: Vztah dat, informací a znalostí (19, s. 8)

### 2.1.2 Informace

Informace jsou jedním z nejcennějších zdrojů v dnešní době. Jsou důležitým faktorem při rozhodování, zmenšují dopad rizika a celkově zjednodušují proces rozhodování. Informace jsou data, kterým jsme schopni porozumět, tedy mají pro nás nějaký smysl. Můžeme je měřit až tehdy, když je převedeme do podoby čísel, a právě ve dvojkové číselné soustavě s nimi pracují veškeré stroje (6).

#### Nejdůležitější faktory informací

Co posuzujeme u informací? Nejdůležitějšími faktory jsou:

- **aktuálnost** (čím aktuálnější informace, tím větší pro nás mají přínos, a to obzvláště v dnešní době, kdy se nacházíme ve velmi turbulentním prostředí),
- **relevanci** (informace je relevantní právě tehdy, když nám přesně odpoví na naši otázku, na to, co jsme potřebovali vědět),
- **pravdivost** (soulad se skutečností, tedy informace bychom měli být schopni nějaký způsobem ověřit) (6).

### 2.1.3 Znalosti

Znalost neboli poznatek je to, co bylo uvedeno do souvislosti, to co jedinec ví poté, co nějaké informace interpretoval v určitém kontextu. Na základě znalostí se může jedinec lépe rozhodovat a zároveň se získané znalosti odráží v jeho jednání. Nositelem znalostí je člověk, který již znalost má. Znalosti můžeme opakovaně používat. Nelze je ovšem řídit tak jako třeba data a informace. Znalosti můžeme rozšiřovat, vyhledávat, ale hlavně aplikovat, což je pro nás nejdůležitější. Právě tak se vyvarujeme následných chyb (6).

## **Základní typy znalostí**

- Explicitní znalosti – jsou to znalosti napsané, vyslovené, můžeme je jednoduše komunikovat, předávat, ukládat
- Implicitní znalosti – jsou založeny na osobní zkušenosti, je těžké je předávat, formalizovat (např. osobní důvěra, pocit bezpečí) (6).

## **2.2 Informační technologie (IT)**

Informační technologií je počítače a veškerá zařízení, která zpracovávají data, informace nebo znalosti. Spadá sem tedy hardware i software, o kterých si řekneme více.

### **2.2.1 Hardware**

Jedná se o veškeré fyzicky existující technické vybavení počítače. Jsou to součástky, bez kterých by počítač nebyl schopen pracovat (např. elektronické součástky, která jsou na základové desce/motherboard/mainboard, procesor, paměť, displej) (8).

### **2.2.2 Software**

Software je veškeré programové vybavení počítače (tedy programy a aplikace v počítači). Software rozdělujeme na aplikační a systémový. Systémový software zajišťuje chod počítače a aplikační software jsou programy, s nimiž pracuje uživatel (7).

## **2.3 Databáze**

Velmi obecně lze databázi definovat jako kolekci souvisejících záznamů, které obsahují vlastní popis. V případě relačních databází (v dnešní době je většina databází relačních) lze definici upravit tak, že se jedná o kolekci souvisejících tabulek. Důležité pro nás

jsou tedy dva výrazy, a to vlastní popis a související tabulky. Související tabulky jsou takové tabulky, které jsou vzájemně propojeny (2).

### **2.3.1 Základní pojmy využívané v databázích**

Pojmy, bez kterých bychom nepochopili tvorbu databází a jejich vlastnosti.

#### **Entita**

*„Entita je prvek reálného světa (např. člověk, stroj, vyučovaný předmět, město), který je popsán svými charakteristikami (vlastnostmi)“ (10).*

#### **Atribut**

Atribut je charakteristika či vlastnost dané entity. Může jí být např.: jméno, příjmení, název, číselná hodnota atd.) (10).

#### **Primární klíč**

Primárním klíčem může být i více atributů než jen jeden. Je to množina atributů, která je jednoznačná (tedy neexistuje v relaci druhá stejná n-tice se stejnými hodnotami) a minimální (tedy ani jeden atribut není možné vypustit, aniž by došlo k porušení jednoznačnosti) (4).

#### **Kandidátní klíč**

Kandidátní klíč je klíč, který má stejné vlastnosti jako klíč primární, ale nebyl vybrán jako primární klíč (4).



## **Cizí klíč**

Je atribut, který je plně zadáný nebo plně nezadáný. Žádná jiná možnost není přístupná. Dále u něj platí, že existuje jiná relace, kde je daný cizí klíč klíčem primárním a má identické hodnoty (4).

## **Doména**

*„Pojmenovaná množina skalárních hodnot stejného typu“ (4).*

## **Normalizace**

Normalizace je proces, kdy se snažíme redukovat opakující se data v databázi. Proces probíhá postupnou dekompozicí relací do vhodnějšího tvaru (5).

Největšími výhodami normalizace jsou:

- snížení opakujících se dat,
- lepší uspořádání a flexibilita databáze,
- lepší zabezpečení databáze (5).

## **První normální forma**

Aby byla tabulka v první normální formě, musí být všechny její atributy entit jednoduché (5).

V praxi jde o to, že každé tabulku rozložíme na co nejjednodušší atributy. Ukážeme si to na příkladu, se složeným atributem adresa. Nyní je například ve formátu „Brno, Nám. 28. října 22, 602 00,“. Aby byla relace v první normální formě, tak potřebujeme atribut adresa rozdělit na jednodušší atributy. Rozdělení provedeme následovně:

Tab. 1: 1. normální forma (vlastní zpracování)

| Město | Ulice          | Číslo po-<br>pisné | PSČ   |
|-------|----------------|--------------------|-------|
| Brno  | Nám. 28. října | 22                 | 60200 |

### **Druhá normální forma**

Cílem druhé normální formy je uložení dat, která jsou částečně závislá na primárním klíči, do jiné tabulky, při dodržení pravidla, že relace je již v první normální formě. Po úpravě do druhé normální formy budou tedy všechny atributy plně závislé na celém primárním klíči (5).

### **Třetí normální forma**

Cílem třetí normální formy je odstranění těch dat, u kterých je funkční závislost na primárním klíči, zprostředkována přes jiný neklíčový atribut (4).

### **Boyce – Coddova normální forma**

Je variací třetí normální formy, kdy musí být zároveň v třetí normální formě (4).

### **Čtvrtá normální forma**

Relace musí být v Boyce-Coddově normální formě a zároveň nesmí být v jedné relaci spojení nezávislé opakované skupiny. Tedy případ, kdy by se opakoval například jeden druh výrobku pouze s rozdílem v množství (4).

### **Pátá normální forma**

*„S pátou normální formou se setkáme pouze okrajově. Týká se případu spojené závislosti, která vyjadřuje cyklické omezení“ (4, s. 62).*

### 2.3.2 Typy databází podle struktury dat

Databáze můžeme dělit podle toho, jakou struktura mají jejich data.

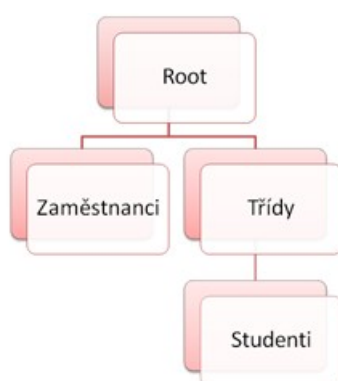
#### Ploché (flat) databáze

Plochá databáze je nejzákladnějším typem databáze. Uchovává data jako obyčejné záznamy. Hlavním rysem takové databáze je, že neobsahuje vazby na jiné záznamy. Příkladem takového typu databáze je i obyčejný textový soubor CSV. Ten na jednotlivých řádcích uchovává jednotlivé záznamy. Využitelnost ploché databáze je optimální v případě nejzákladnějšího ukládání záznamů (9).

#### Hierarchické a síťové databáze

Hierarchické a síťové databáze se již dnes moc nepoužívají. Jsou brány spíše jako předchůdci dnes využívaných databází (9).

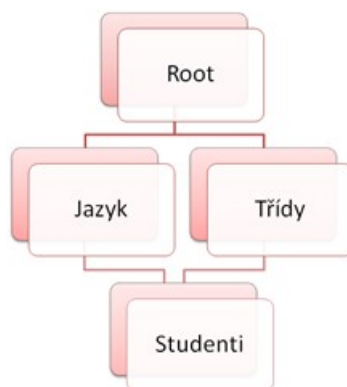
Vyznačují se tím, že oba typy mají snahu o eliminaci redundantních dat. Dovolují definovat k jednotlivým tabulkám rodičovské tabulky. Platí pak, že každý záznam v tabulce musí odkazovat na jeden ze záznamů z hierarchicky vyšší tabulky (9).



Obr. 2: Hierarchická databáze (9)

Na obrázku vidíme příklad hierarchické databáze, kdy každý student musí mít odkaz na jeden záznam z tabulky „Třídy“ a zabírá tím minimum prostoru. Tím je omezena redundance dat (9).

Síťová databáze má oproti hierarchické navíc jedno rozšíření, a tou je možnost více rodičovských tabulek (9).



Obr. 3: Síťová databáze (9)

Tedy je možné, aby si každý student zvolil jeden povinný jazyk a chodil do jedné třídy. Ovšem je jasné, že takový model nám nestačí. Proto jsou dnes nejpoužívanějším typem databáze relační, o kterých si dál něco povíme (9).

## Relační databáze

Relační databáze nemají nedostatky předešlých typů databází. Nahlíží na jednotlivé tabulky s daty jako na tabulky na stejné úrovni. Můžou obsahovat libovolný počet tabulek, které jsou k ukládání dat. Data se ukládají do řádků podle předem definovaných pravidel sloupců tabulky (9). Příkladem může být tato tabulka:

Tab. 2: JEDNOTKY (vlastní zpracování)

| id_jednotky(PK) | nazev    | id_zkratky(FK) |
|-----------------|----------|----------------|
| 1               | kilogram | 1              |
| 2               | kus      | 2              |

Každá tabulka musí mít primární klíč. Primární klíč je nějaký jednoznačný identifikátor záznamu (sloupec, který musí obsahovat unikátní hodnotu v rámci všech řádků tabulky), zde jím je `id_jednotky`. Dále pak mohou mít i cizí klíče, ty definují odkaz na primární klíč z jiné tabulky. Díky tomu pak můžeme odkazovat jeden řádek tabulky na řádek tabulky jiné (9).

Tab. 3: ZKRATKY (vlastní zpracování)

| <code>id_zkratky</code> (PK) | <code>zkratka</code> |
|------------------------------|----------------------|
| 1                            | kg                   |
| 2                            | ks                   |

## **Integritní relačního modelu**

Integrita je stav, kdy data modelu reprezentují vlastnosti objektů běžného světa. Tyto integritní omezení může rozlišovat na integritní omezení pro entity a integritní omezení pro vztahy entit (4).

### **Integritní omezení pro entity**

Známe integritu doménovou, entitní a referenční (4).

### **Doménová integrita**

Doménové integritě se také říká integrita hodnot, protože omezuje hodnoty atributů relace. Povoluje určité hodnoty danému atributu jako třeba datový typ, jedinečnost ve sloupci dané tabulky, rozsah hodnot a další (4).

### **Entitní integrita**

Entitní integrita říká, že každá relace musí mít určený primární klíč (4).

## Referenční integrita

O referenční integritě mluvíme, když nám cizí klíč spolu s primárním klíčem jiné tabulky vytváří spojení mezi relacemi. Jsou zde ale dvě pravidla. Cizí klíč s primárním klíčem, který mu odpovídá, jsou definovány na stejné doméně. Druhým pravidlem je, že se v databázi nesmí nacházet jakákoliv nesouhlasná hodnota cizího klíče (4).

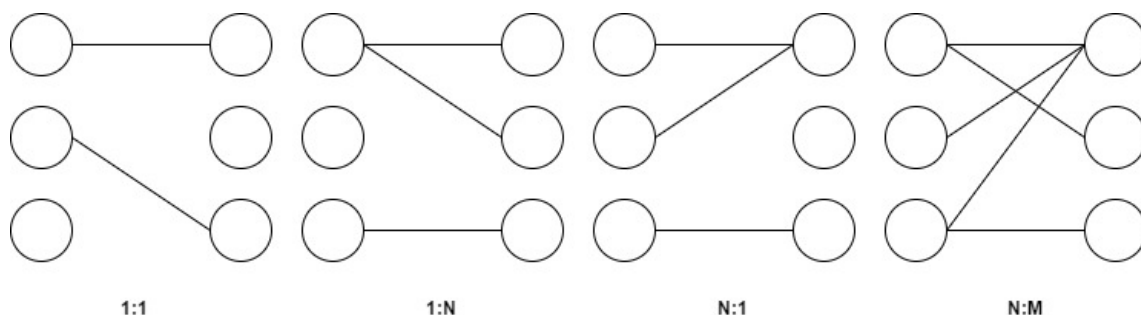
## Integritní omezení pro vztahy entit

Protože entity jsou prvky reálného světa, mají mezi sebou stejně tak určité vztahy. Tyhle vztahy se dělíme na čtyři typy. Prvním typem vztahu je 1:1, kdy příkladem je, když jeden člověk má jedno rodné číslo a jedno konkrétní rodné číslo může mít pouze jeden člověk (4).

Opakem je vztah typu N:1, kdy příkladem může být, když více lidí používá jedno auto (4).

Dalším typem vztahu je 1:N, kdy například jeden člověk může mít více kreditních karet, ale konkrétní karty může mít jen jeden člověk, majitel karet (4).

Posledním typem vztahu je pak M:N, kdy příkladem může být reálný vztah, kdy jeden člověk může chodit na více kurzů a zároveň do jednoho kurzu může chodit i více lidí (4).



Obr. 4: Vazby mezi entitami (4, s. 30)

## Objektové databáze

Objektové databáze namísto relací ukládají přímo objekty, včetně jejich vlastností. Místo řádků tabulek se zde ukládají samotné instance objektů. Každý objekt můžeme snadno a jednoznačně identifikovat podle jeho OID (Object ID). Není tedy potřeba vytvářet primární klíče na objektech a ani není potřeba normalizovat tabulky (11).

Objektové databáze také nabízejí využití možností vícenásobné dědičnosti, zapouzdření a polymorfizmu. Zapouzdření znamená, že každý objekt obsahuje data i funkce, které říkají, jak je možné s vlastnostmi objektu zacházet. Polymorfizmu je pak možnost zastupovat své potomky v případě volání metod (11).

## 2.4 Databázové systémy

Databázový systém zahrnuje 4 komponenty: systém řízení databáze, vlastní databázi, databázovou aplikaci a uživatele.

### 2.4.1 Systém řízení databáze (DBMS – Database management system)

Systém řízení databáze je velký a složitý produkt, který se většinou pořizuje formou licence od dodavatele softwaru. Hlavním úkolem systému řízení databáze je vytváření, zpracování a spravování databáze. Dr. Edgar Frank Codd definoval 12 pravidel, které mají splňovat databázové systémy založené na relačním modelu.

*„Jsou to:*

- **Informační pravidlo**

*informace v relační databázi jsou na logické úrovni vyjádřeny hodnotami v tabulkách,*

- **Pravidlo jistoty**

*všechna data v relační databázi jsou zaručeně přístupná,*

- **Systematické zpracování nulových hodnot**

*nulové hodnoty jsou plně podporovány při zobrazování, a to nezávisle na datovém typu,*

- **Dynamický on-line katalog založený na relačním modelu**

*popis databáze je vyjádřen na logické úrovni stejným způsobem jako zákaznická data, takže autorizovaný uživatel může aplikovat stejný relační jazyk ke svému dotazu jako uživatel při práci s daty,*

- **Obsáhlý datový podjazyk**

*relační systém může podporovat několik jazyků a různých módů použitých při provozu terminálu. Nicméně musí být nejméně jeden příkazový jazyk s dobře definovanou syntaxí, který obsáhle podporuje definici dat, definici pohledů, manipulaci s daty jak interaktivně, tak programem, integritní omezení, autorizovaný přístup k databázi, transakční příkazy apod.,*

- **Pravidlo vytvoření pohledů**

*všechny pohledy, které jsou teoreticky možné, jsou také systémem vytvořitelné,*

- **Schopnost vkládání, vytvoření a mazání**

*schopnost zachování relačních pravidel u základních i odvozených relací je zachována nejen při pohledu na data, ale i při operacích průniku, přidání a mazání dat,*

- **Fyzická datová nezávislost**

*aplikační programy jsou nezávislé na fyzické datové struktuře,*

- **Logická datová nezávislost**

*aplikační programy jsou nezávislé na změnách v logické struktuře databázového souboru,*

- **Integritní nezávislost**

*integritní omezení se musí dát definovat prostředky relační databáze nebo jejím jazykem a musí být schopna uložení v katalogu, a nikoliv v aplikačním programu,*

- **Nezávislost distribuce**

*relační SŘBD musí být schopny implementace na jiných počítačových architekturách,*



- ***Pravidlo přístupu do databáze***

*jestliže má relační systém jazyk nízké úrovně, pak tato úroveň nemůže být použita k vytváření integritních omezení a je nutno vyjádřit se v relačním jazyce vyšší úrovně“(12).*

## **2.4.2 Uživatel**

Uživatel pomocí databázové aplikace udržuje přehled o vybraných skutečnostech, může pomocí formulářů načítat, zadávat data a dotazovat se na ně. Případně si může vytvářet sestavy (2).

## **2.4.3 Databázová aplikace**

Databázové aplikace jsou sady jednoho nebo více počítačových programů, které jsou něco jako prostředník mezi uživatelem a systémem řízení databáze. Čtou nebo upravují databázová data (2).

## **2.4.4 Vlastní databázi**

Jak již víme, databáze je zjednodušeně kolekce nějakých souvislých záznamů.

## **2.5 Jazyk SQL**

Jazyk SQL je jazyk neprocedurální, to znamená, že nepopisuje, jak se má něco provádět, ale co se má provádět, na jakých datech se má pracovat.

Mámě dvě standardizační organizace, které propagují jazyk SQL jako průmyslový standard. První organizací je American National Standards Organization (standard ANSI), druhou je International Standards Organization (standard ISO) (2).

### **2.5.1 Historie jazyka SQL**

Historie jazyka SQL začíná v laboratořích společnosti IMB, v San Jose, v Kalifornii, kde byl jazyk vyvinut. Zkratka SQL znamená Structured Query Language (strukturovaný dotazovací jazyk). Původně měl být jazyk pro produkt DB2 od již zmíněné společnosti IBM, což je relační databázový systém, který by se v dané době bez jazyka SQL nemohl existovat (2).

## **2.6 Vývojové prostředí MS SQL Server 2012**

Historie SQL server začala v roce 1988, kdy neměla nic společného s Microsoftem. Byl to výtvar společnosti Sybase. Sybase v roce 1993 uvedla klasickou desktopovou verzi, která byla určena pro kanceláře a malé firmy s operačním systémem Windows. Až v roce 1994 dochází ke koupi SQL Serveru Microsoftem, který ho vyvinul do formy, jak ho známe dnes (2).

### 3 ANALÝZA SOUČASNÉHO STAVU

V této kapitole si provedeme analýzu současného stavu elektronického obchodu. Uvedeme si základní informace o firmě, dále pak organizační strukturu, používaný software a hardware a jeho omezení, zanalyzujeme si konkurenci a nezapomeneme i na naše zákazníky.

#### 3.1 Základní údaje o firmě

**Název:** Tomáš Kadleček

**Právní forma:** Fyzická osoba

**Sídlo:** Gellnerova 95/10a, Brno-Jundrov, 637 00

**IČ:** 04762185

**Kontakt:** +420 739 667 558

**Web e-shopu:** [www.wowdoplňky.cz](http://www.wowdoplňky.cz)

**Velikost firmy podle počtu zaměstnanců:** Mikropodnik



Obr. 5: Logo e-shopu (vlastní zpracování)

### 3.2 Hlavní činnost

E-shop WOWdoplňky.cz se zabývá především prodejem módních doplňků, jako jsou kravaty, motýlky, manžetové knoflíčky, kapesníčky do saka, spony na kravaty a spousta dalších, dále pak prodejem společenského oblečení.

### 3.3 Historie

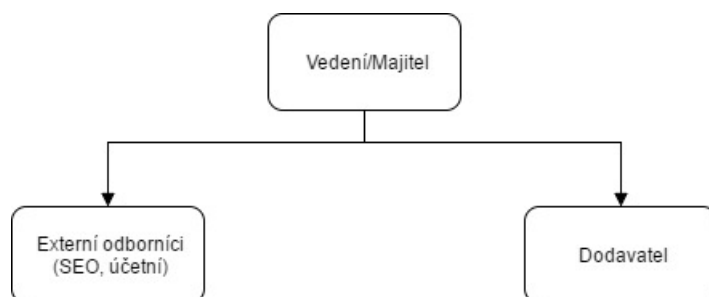
E-shop byl založen začátkem roku 2016, kdy e-shop získal dotaci ve výši 80 000 Kč od Jihomoravského kraje v rámci jejich dotačního programu pro začínající podnikatele. Cílem bylo vytvořit funkční e-shop s minimální potřebou investování velkých sum do skladových zásob.

Důležité je zmínit, že e-shop WOWdoplňky.cz dosud nemá žádnou kamennou pobočku.

### 3.4 Organizační struktura

O e-shop se stará pouze jeho majitel. Má na starost vše od komunikace se zákazníkem, dodavatelem, objednáváním zboží až po marketing.

Složitější úpravy e-shopu se řeší externě ve spolupráci s jedním programátorem. Dále pak máme člověka pro problematiku SEO.



Obr. 6: Organizační struktura (vlastní zpracování)

### **3.5 Konkurence**

V dnešní době je na trhu desítky e-shopů, které prodávají oblečení. Proto byl původní cíl, mít e-shop se zaměřením spíše na módní doplňky.

V ČR fungovalo v roce 2015 okolo 36 800 e-shopů, tedy je jasné, že konkurence je obrovská. Největšími konkurenty v ČR jsou e-shopy jako ZOOT, Wayfarer, Vkosili, ale i spousta dalších.

Hlavní snahou e-shopu je rozšíření nabízeného sortimentu. Nyní e-shop nabízí necelé 2000 modelů zboží. V plánu je rozšíření nabízeného sortimentu na 5000 modelů a personálu. Dalším cílem pro tento rok je postupné rozšiřování popisků zboží, které je velmi důležité z hlediska zlepšení pozic ve organickém vyhledávání. Splnění zmíněných cílů by pomohlo ke zvýšení konkurenceschopnosti e-shopu.

### **3.6 Zákazníci**

E-shop se snaží cílit na Českou republiku a Slovensko. Odhadem tak získává okolo 6 milionů potenciálních zákazníků. Pro každý e-shop v dnešní době je důležité, dokázat potenciální zákazníky správně oslovit, což jde skvěle pomocí online reklamy, kde můžeme cílit podle zájmů, pohlaví, věku, ale i podle podobnosti zákazníků a spousty dalších kritérií. Další důležitou věcí je budovat základnu spokojených zákazníků, kteří se budou pravidelně vracet. Proto je potřeba, se o zákazníky dobře starat. Příkladem může být budování online poradny pro nákup, kvalitní a rychlá komunikace v průběhu nákupního procesu, ale i po něm.

### **3.7 Hardware**

E-shop disponuje jedním notebookem, jedním pevným PC a pak laserovou tiskárnou pro tisk faktur zákazníkům a jejich další evidenci. Dále pak dvěma Wi-Fi routery pro internetové připojení.

## **3.8 Software**

### **3.8.1 Operační systém**

Jak notebook, tak i pevný počítač běží na operačním systému Windows 10. Windows 10 byly uvedeny na trh roku 2015 a jsou pokračováním verze Windows 8. Viditelný rozdíl je v tom, že by měli být uživatelsky přívětivější. Je zde i spousta vylepšení, které objevíme, když se budeme o Windows 10 zabývat podrobněji. Windows 10 podporují veškeré programy potřebné pro práci na e-shopu.

### **Shoptet platforma**

E-shop běží na platformě Shoptetu, vytvořila ji stejnojmenná česká firma v čele s Miroslavem Ud'anem, která patří mezi jedny z největších poskytovatelů e-shopových řešení na českém trhu. Na platformě Shoptetu funguje přes 8300 e-shopů, podle statistik z roku 2015. Firma má okolo 40 zaměstnanců (13).

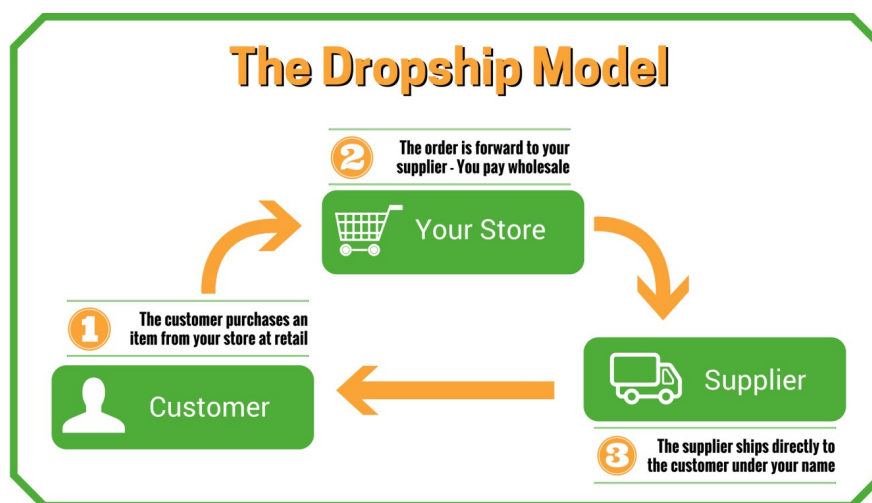
Jedním z nejdůležitějších faktorů, proč je e-shop na platformě Shoptetu je, že má platforma silnou zákaznickou podporu a oddělení vývoje, a tak dochází k pravidelnému vylepšování jejích funkcí, vzhledu a možností.

Výhodou platformy pro uživatele je, že se nemusí nijak instalovat či implementovat. Vše totiž běží na webu Shoptetu, kde se každý správce pomocí přihlašovacího rozhraní přihlásí do své administrace. Odpadají tak zároveň vysoké požadavky na úložiště.

### **Dodej zboží**

E-shop funguje na bázi dropshippingu. To znamená, že veškeré zboží, co nabízí, není na jeho skladě a tím pádem ani není nutné investovat do skladových zásob. Výjimkou jsou vratky zboží, které e-shop už nemůže vrátit dodavateli. Celý proces dropshippingu funguje tak, že zákazník na e-shopu udělá objednávku a zaplatí, e-shop pak předá objednávku dál na dodavatele, který zboží připraví a dále pak expeduje přímo ke koncovému zákazníkovi pod jménem e-shopu.

S dodavatelem je uzavřena smlouva o spolupráci. Dodavatel tak prodává e-shopu zboží za velkoobchodní ceny, tedy levněji, než by mohl nakupovat klasický zákazník. Je to zapříčiněné tím, že e-shop dodavateli zajistí určitý odbyt zboží, a to bez jakékoliv další investice do inzerce, kterou by za normálních okolností musel vynaložit.



Obr. 7: The Dropship Model (19)

### 3.9 Základní procesy stávajícího řešení

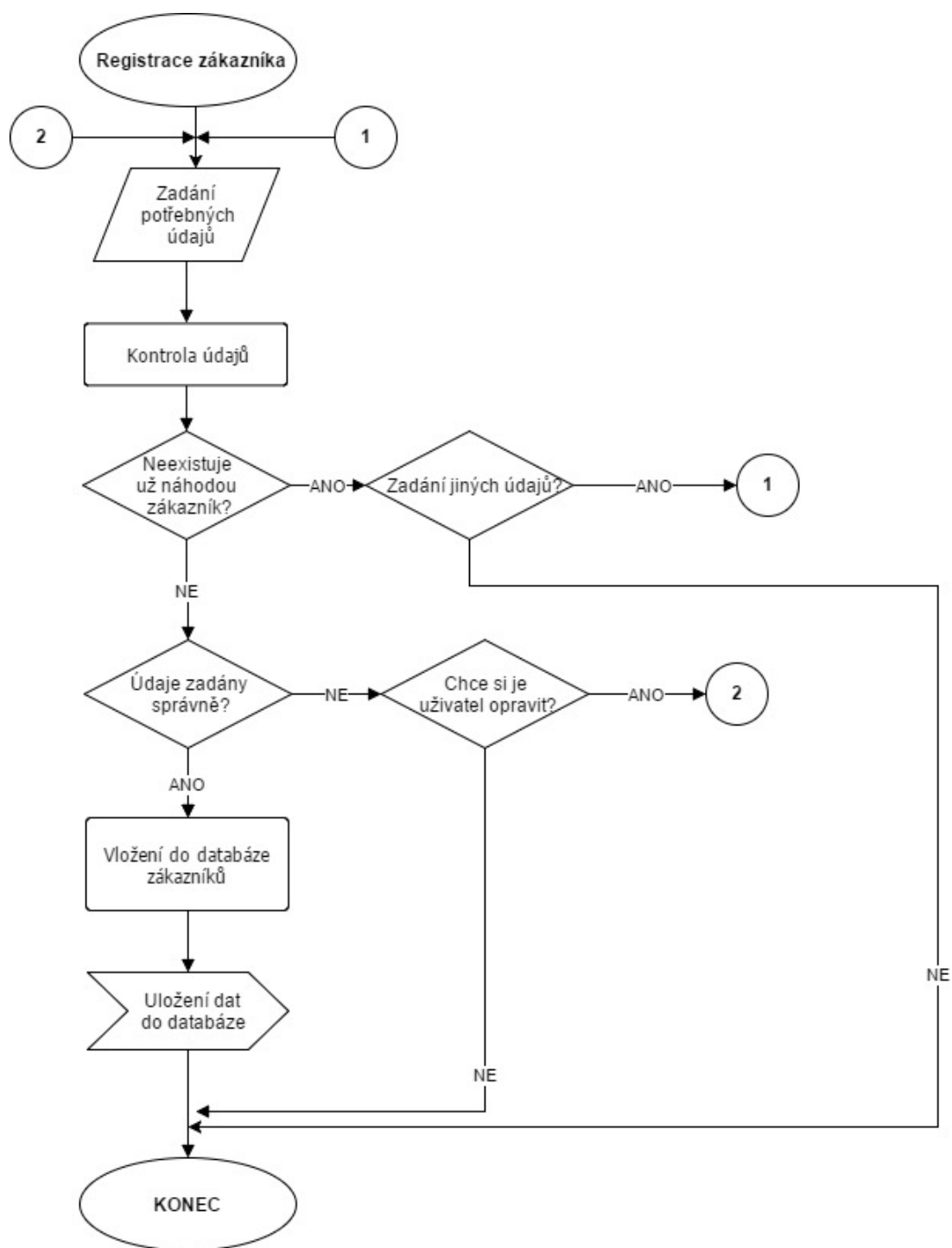
Základními procesy, které byly identifikovány na základě zkušeností se stávajícím e-shopem jsou: Registrace a přihlášení zákazníka, Vytvoření objednávky (s kontrolou dostupnosti zboží), aktualizace dat.

#### 3.9.1 Registrace zákazníka

Registrace zákazníka začíná zadáním osobních a kontaktních údajů potřebných k registraci. K těmhle údajům patří i volba přihlašovací email a heslo. Již při zadávání údajů dochází ke kontrole, zda už takový zákazník neexistuje. V případě že existuje, dojde k výzvě o zadání nových údajů. V opačném případě dochází ke kontrole, zda jsou údaje zadány správně, tedy například, zda je email ve správném formátu nebo jestli heslo spl-

ňuje všechny požadavky. Při správnosti údajů proces pokračuje vložním zákazníka do databáze. Při nalezení chyby v údajích dochází k výzvě o opravu daného údaje. Proces končí uložením zákazníka do databáze.





Obr. 8: Registrace zákazníka (vlastní zpracování)

## Nedostatky procesu registrace zákazníka

V Shoptetu bohužel chybí, možnost členit zákazníky při registraci podle určitých kritérií, která jsou pro e-shop důležitá. Jedinou možností je, zde vytvořit takovou databázi ručně, kde je hlavním problémem časová náročnost.

O zákaznících má smysl zaznamenávat následující údaje:

- **jméno a příjmení** (kvůli personalizaci marketingu),
- **datum narození** (možnost použití slev na narozeniny),
- **e-mail** (pro zasílání newsletteru, pro email marketing),
- **místo bydliště** (pokud e-shop má místa odběru, pak lze použít remarketing podle místa lokace),
- **hodnota všech nákupů** (možnost využití věrnostního programu, při určité útratě přidělit zákazníkovi věrnostní slevu),
- **věk** (lze použít pro segmentaci zákazníků).

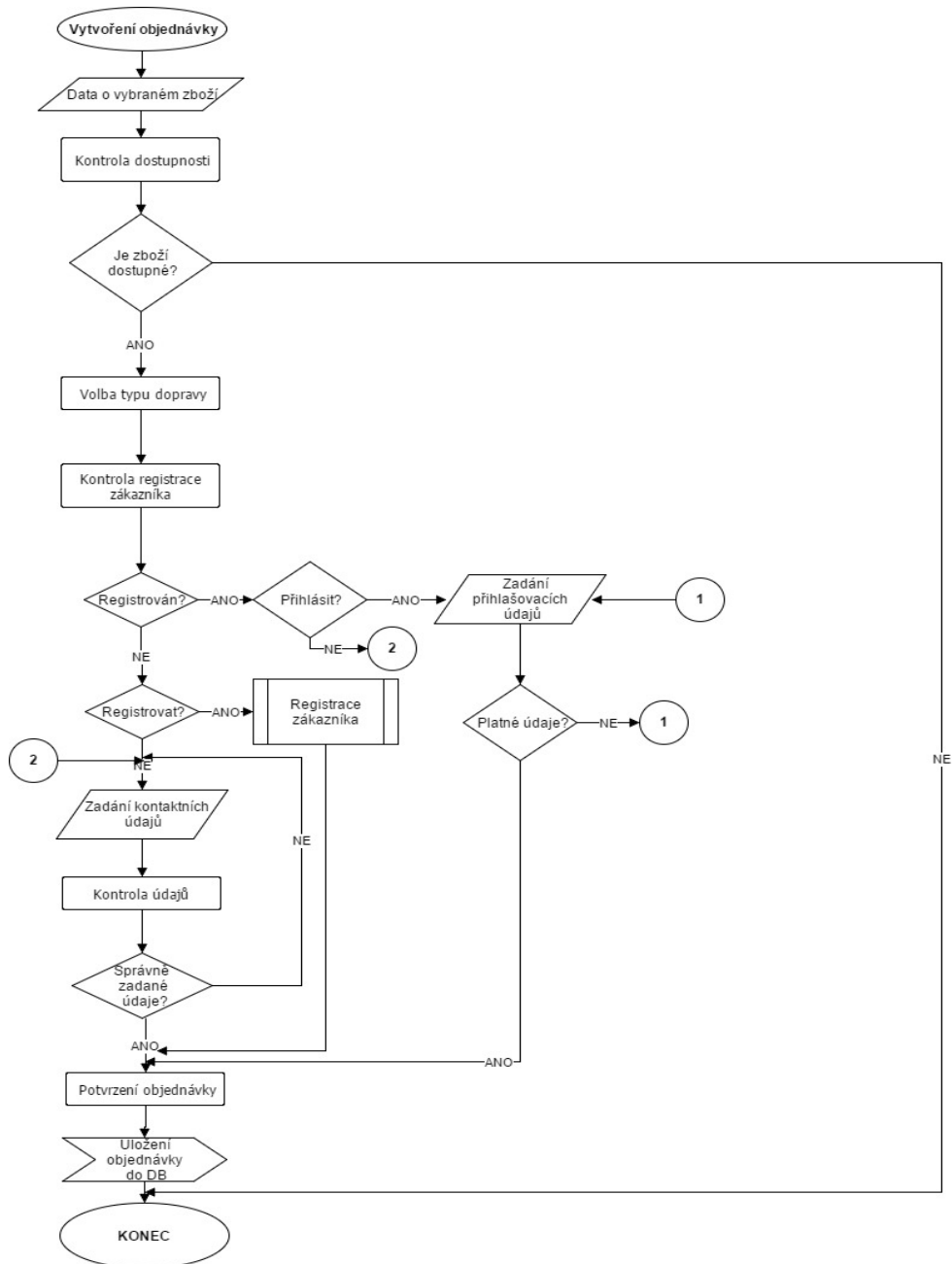
V souhrnu je taková databáze o zákaznících základ pro další práci se zákazníky, především však pro prodej.

### 3.9.2 Vytvoření objednávky

Zde dochází k vytvoření objednávky zákazníkem. Zákazník si vybere zboží a dochází ke kontrole dostupnosti. Nejprve si zvolí, jestli se chce zaregistrovat (nebo přihlásit) či pokračovat bez registrace. V případě přihlášení má zákazník již předvyplněné osobní a kontaktní údaje. V případě registrace dochází ke stejnému postupu, jako je proces registrace. V případě vytvoření objednávky bez registrace musí zákazník, vyplnit osobní a kontaktní údaje. Dalším krokem je volba typu dopravy a platby. Zákazník si zde vybere ze dvou druhů dopravy (GLS, Česká pošta) a dvou druhů platby (Dobírka, platba pře-

vodem na bankovní účet). Posledním krokem je kontrola vyplněných údajů a následně potvrzení objednávky.

Dále se odešle objednávka dodavateli, který ji připraví a odešle přímo zákazníkovi.



Obr. 9: Vytvoření objednávky (vlastní zpracování)

### **Nedostatky procesu vytvoření objednávky**

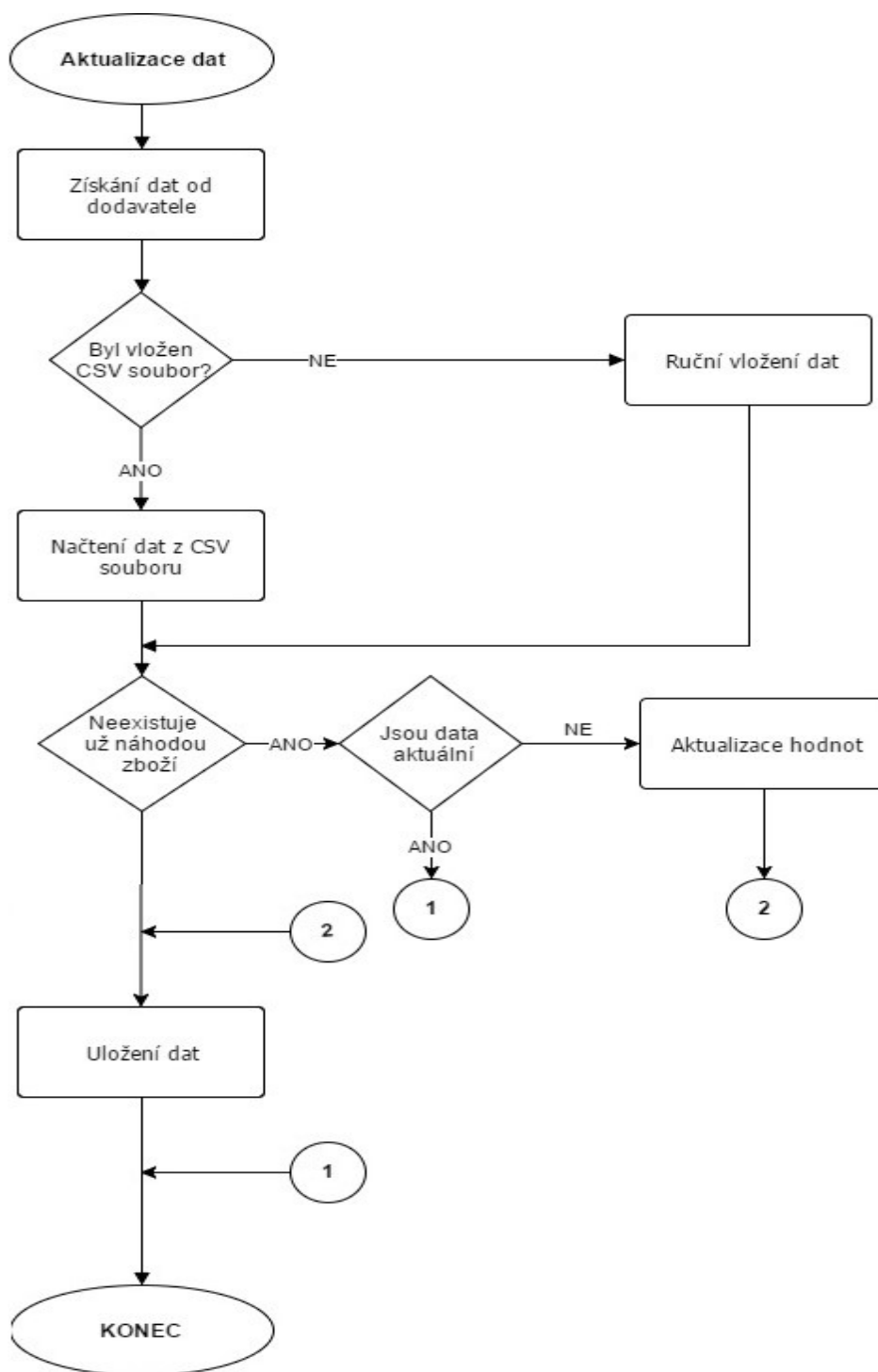
V případě vratek od zákazníků se na skladě může hromadit vrácené zboží a e-shop na tom zbytečně trátí. PŘ: Při dalších objednávkách si zákazník objedná zboží, které má e-shop dostupný jako vratku, ale není schopen správně zareagovat a místo expedování zboží z „Vratek“ dojde k nové objednávce zboží od dodavatele.

### **3.9.3 Proces aktualizace dat**

Aktualizace dat funguje v nynějším řešení e-shopu pomocí importu dat ve formě CSV souboru od dodavatele nebo ručně, kdy aktualizovaná data získává od dodavatele. Dojde tedy ke kontrole, zda byl vložen CSV soubor. Pokud ano, následuje načtení CSV souboru. Pokud ne, je vyžadováno ruční vložení dat. Po vložení dat dochází ke kontrole, zda zboží již existuje nebo ne. Existuje-li, zajímá nás, jestli jsou data aktuální. V případě neaktuálních dat dojde k aktualizaci. Posledním krokem je uložení dat.

### **Nedostatky procesu aktualizace dat**

Tohle řešení má značné nedostatky jako téměř nulová automaticnost a časová náročnost aktualizace.



Obr. 10: Aktualizace dat (vlastní zpracování)

## 4 NÁVRH VLASTNÍHO ŘEŠENÍ

V této části se budeme zabývat vlastním návrhem na řešení databáze. Nejprve si popíšeme základní procesy v databázi a stanovíme si požadavky na databázi. Poté, provedeme konceptuální (jehož součástí bude i definice základních entit), logický i fyzický návrh. Ke konci si řekneme, proč jsem pro práci zvolil nástroj SQL Server 2014 Management Studio a úplně nakonec se zaměříme na zhodnocení přínosů navržené databáze.

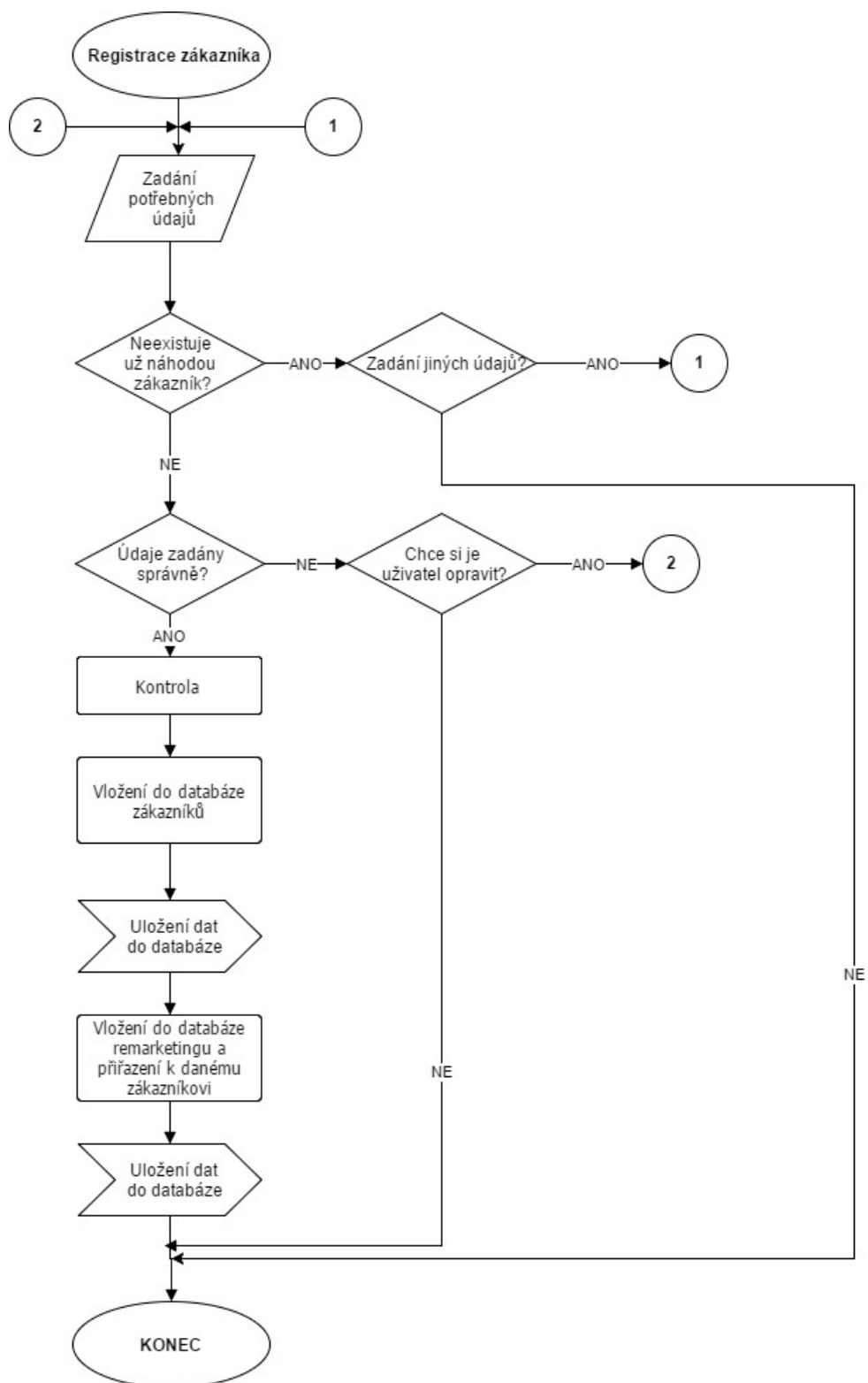
### 4.1 Základní procesy v databázi

Zde si definujeme důležité procesy a uvedeme si rozdíly oproti původnímu řešení.

#### 4.1.1 Registrace a přihlášení zákazníka

V případě nového zákazníka, který se chce registrovat, musí nejprve dojít k výzvě o zadání osobních a kontaktních údajů jako jsou např. jméno, příjmení, e-mail, login a heslo. Heslo by mělo být zadáváno ve dvou krocích, tedy zadání hesla poprvé a zadání hesla podruhé pro ověření správnosti. Dále je třeba při registraci kontrolovat email, a to ať už jeho formát či již nabytou existenci.

Dochází tedy ke kontrole správnosti údajů. V případě již existujícího uživatele/zákazníka dochází k výzvě o opravu či zadání zcela nových údajů. Po úspěšném zadání údajů dochází k zapsání uživatele do databáze zákazníků i remarketingu a následně pak k uložení údajů.



Obr. 11: Registrace zákazníka - nově (vlastní zpracování)

## **Rozdíl oproti původnímu řešení**

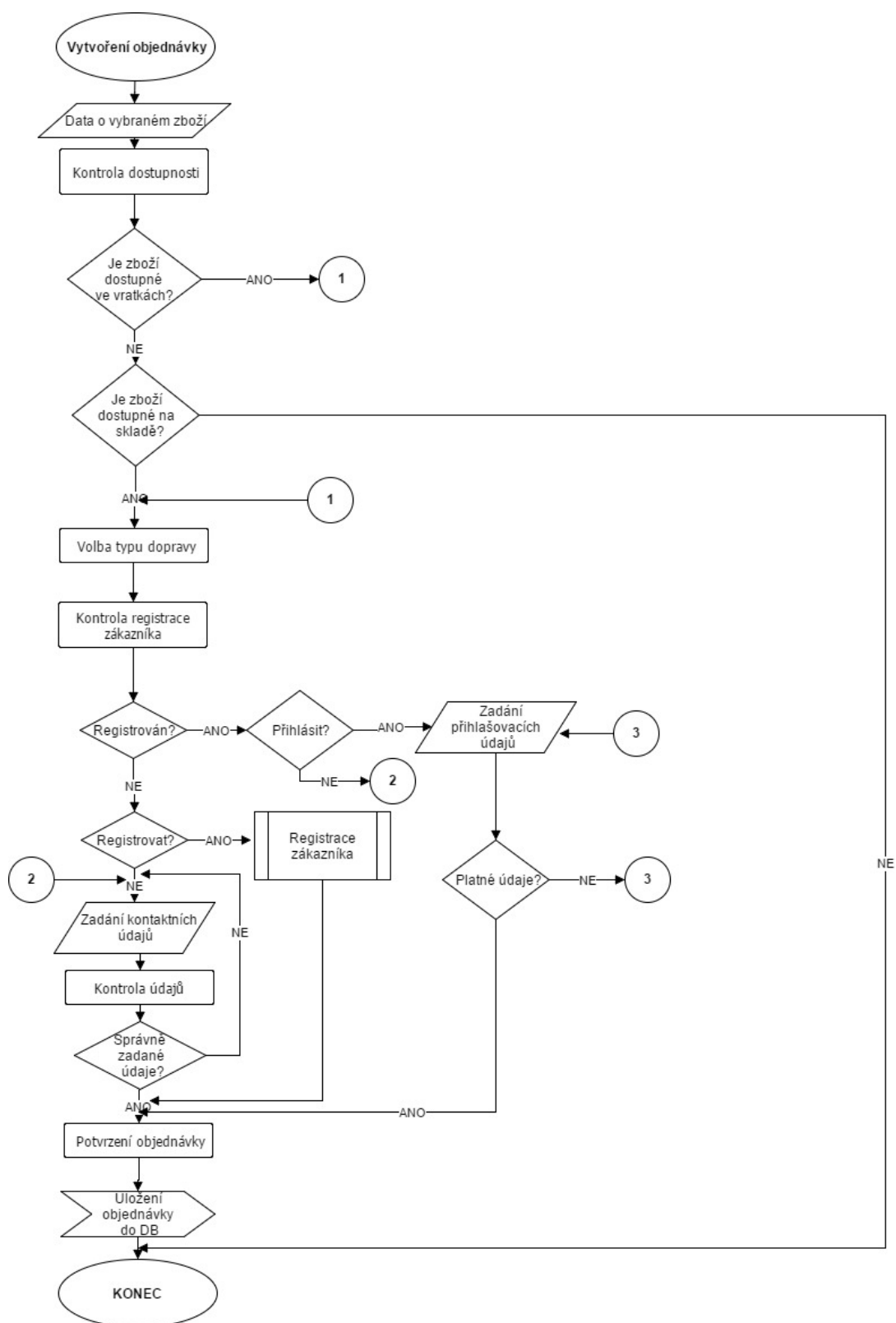
Rozdílem zde je krok uložení zákazníka do databáze remarketingu. Tedy při registraci zákazníka dochází automaticky k vytvoření záznamu o zákazníkovi i v tabulce s remarketingovými daty. Tabulka se pak naplňuje daty, která e-shopu pomohou zákazníky segmentovat. Můžou to být data s přehledem o návštěvách webu zákazníkem nebo informace, zda daný zákazník vložil nějaký produkt do košíku v posledních 30 dnech. Taková data pak mohou říct, jak moc je zákazník připravený k nákupu.

### **4.1.2 Vytvoření objednávky**

Zde dochází k vytvoření objednávky zákazníkem. Zákazník si nejprve vybere zboží, pak následuje kontrola dostupnosti zboží. Prvně dochází ke kontrole, zda se zboží nenachází ve vratkách, aby nedocházelo k zbytečnému objednávání od dodavatele. V případě že zde zboží není, dochází k objednání od dodavatele.

Proces vytvoření objednávky pokračuje volbou zákazníka, zda se chce zaregistrovat (nebo přihlásit) či pokračovat bez registrace. V případě přihlášení má zákazník již předvyplněné osobní a kontaktní údaje. V případě registrace dochází ke stejnému postupu, jako je proces „Registrace zákazníka“. V případě vytvoření objednávky bez registrace musí zákazník vyplnit osobní a kontaktní údaje. Dalším krokem je volba typu dopravy a platby. Zákazník si zde vybere ze dvou druhů dopravy (GLS, Česká pošta) a dvou druhů platby (Dobírka, platba převodem na bankovní účet). Posledním krokem je kontrola vyplněných údajů a následně potvrzení objednávky.





Obr. 12: Vytvoření objednávky - nově (vlastní zpracování)

## **Rozdíl oproti původnímu řešení**

Rozdílem je evidence vratek od zákazníků. Tedy při objednávce dochází nejprve ke kontrole, zda není objednávané zboží ve vratkách, až poté dochází ke kontrole dostupnosti na skladě. V případě nalezení zboží ve vratkách, dochází k expedici zboží z vratek, a ne ze skladu. Toto řešení tedy dokáže šetřit e-shopu náklady, které by jinak zaplatil za objednání nového zboží a případně i dopravy.

### **4.1.3 Návrh na změnu procesu aktualizace dat**

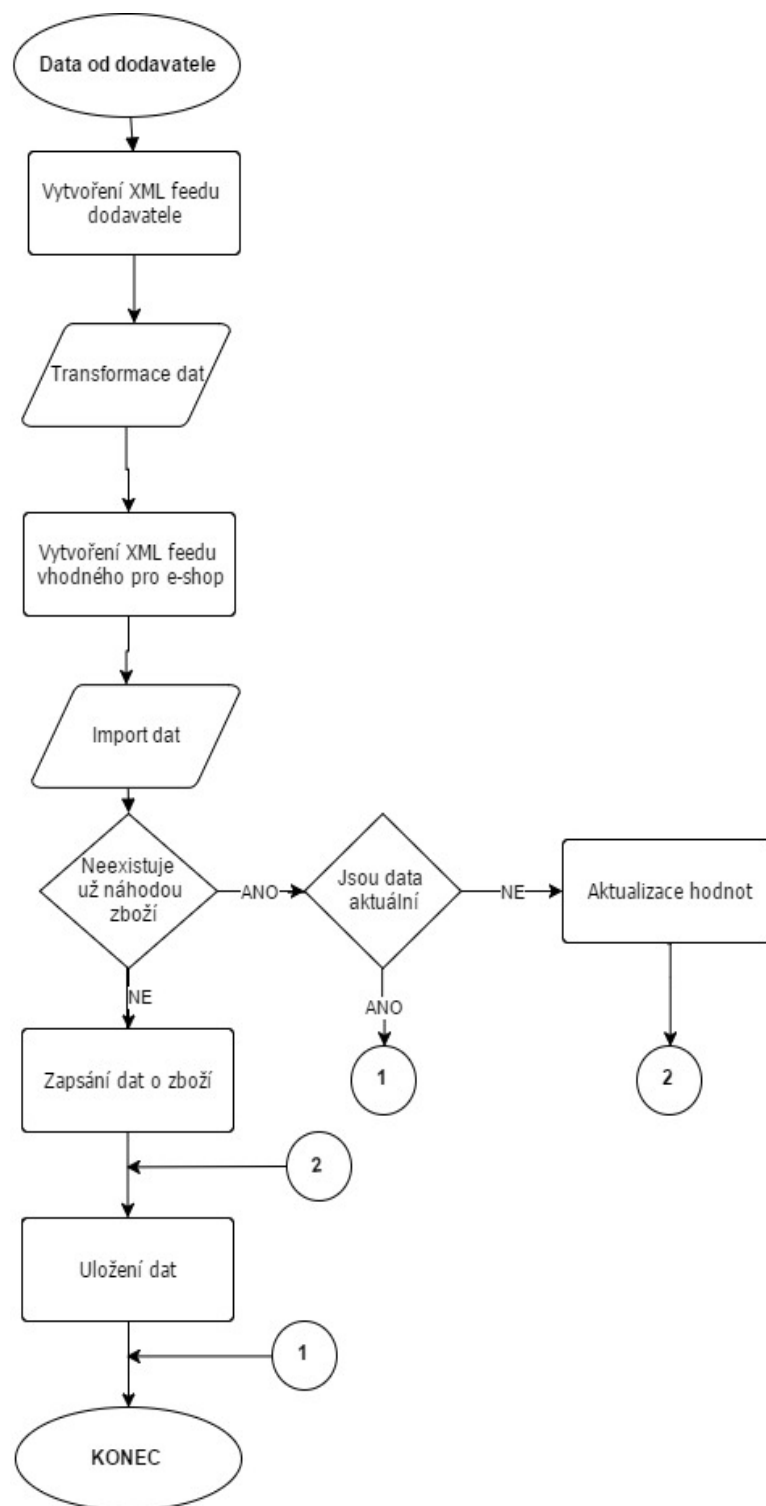
Jsem názoru, že by bylo lepší využít automatické aktualizace dat. Taková změna dokáže e-shopu ušetřit hodiny a hodiny času každý týden. Zbude tak více času, který může být investován lépe.

Celý proces začíná transformací dat XML feedu dodavatele. Součástí XML feedu od dodavatele jsou data jako název produktu, ceny, množství, kategorie, do které se má produkt importovat, dále pak třeba materiál, barvu, délku garance a jiné. Dochází k transformaci dat do podoby, kterou dokáže e-shop zpracovat. V téhle podobě se XML feed uloží na námi zvolenou URL adresu. My pak následně z dané URL adresy stahujeme data do databáze e-shopu.

Dojde tedy k příjmu dat. Pokud se nějaký produkt v XML feedu nenachází, pak se mu do proměnné množství, přiřadí hodnota 0. V takovém případě dochází i ke skrytí produktu v navigaci e-shopu. Díky tomu nedochází k objednávkám, které nejsme schopni realizovat. Pokud se produkt v XML feedu nachází, tak je pomocí jednoznačného identifikátoru spárován a aktualizován.

## **Rozdíl oproti původnímu řešení**

Rozdílem nového řešení od původního je, že k aktualizaci dat dochází pomocí automatického importu. Tedy není nutné data aktualizovat ručně. Data se aktualizují dle nastavení. (1x denně, 3xdenně, 1x za hodinu...). Dochází tak k automatizaci procesu aktualizace zboží a dat o zboží.



Obr. 13: Aktualizace dat - nově (vlastní zpracování)

## **4.2 Požadavky na databázi**

Základním požadavkem je navrhnout databázi pro elektronický obchod. E-shop musí být schopný data načítat, ukládat, upravovat nebo i vytvářet.

Požadavky na databázi byly stanoveny na základě vlastních zkušeností s fungováním e-shopu a platformy, na které e-shop funguje.

### **4.2.1 Dílčí požadavky na databázi**

Zde si uvedeme dílčí požadavky, se kterými je potřeba při návrhu počítat.

#### **Jednoduchost**

Důležité je, aby byla databáze co nejjednodušší a tím pádem se s ní dobře pracovalo.

#### **Aktuálnost dat**

Aby e-shop fungoval, jak má a dokázal správně reagovat na objednávky, je potřeba mít data aktuální. Toho docílíme každodenní aktualizací dat.

#### **Přehled všech údajů o zákaznících**

Přehled údajů o zákaznících pro následné užítí například k online marketingu.

#### **Rychlost**

Databáze musí být optimalizovaná, aby nedocházelo ke zpoždění odezev.

## **Příprava pro EET**

V dnešní době, v roce 2017 je aktuálním problémem elektronická evidence tržeb. Databáze byla proto navržena tak, aby byla připravena pro tisk EET účtenek.

## **Sbírání dat o zákaznících**

Při registraci nebo objednávce nového zákazníka musí docházet k přidání údajů o zákazníkovi do tabulky remarketingu. Pokud zatrhl možnost odebírat newsletter, pak dojde i k označení, že newsletter povolil a díky tomu můžeme následně používat emailing.

## **Uživatelé databáze**

Databáze je určená pro zaměstnance a má jim zjednodušit práci s údaji o zákaznících. Má pomoci s přípravou pro remarketing a e-mailing. Pro zaměstnance to bude velká úspora času a zároveň tak dojde k částečné automatizaci některých činností a tím i snížení nákladů.

## **Zákazníci**

Každý zákazník by měl mít přístup ke svým osobním údajům a možnosti jejich úprav. Registrovaní zákazníci pak mohou využívat i věrnostních slev, které jim budou uděleny při dosažení určité hodnoty všech jejich dosavadních objednávek.

## **Zboží**

Nutností je možnost data o zboží automaticky importovat. Pokud bychom měli všechny aktualizace řešit ručně, tak bychom nebyli schopní včas reagovat na změny. Docházelo by pak k mylným objednávkám a zvyšovala by se tak i nespokojenost zákazníků se službami.

### **Bezpečnost (z hlediska zákazníků)**

Šifrování a skrytí hesel zákazníků, aby nedošlo k zneužití dat někým cizím. Tedy v případě zadávání hesla při přihlašování je nutné heslo skrýt, aby nedošlo k phishingu či jiným podvodným metodám. Dále pak při ukládání databáze musí dojít k zašifrování hesla. Heslo by tak nemělo být možné získat v nezabezpečené podobě.

### **Bezpečnost (z hlediska zaměstnanců)**

Databáze musí být zabezpečena heslem. Každý zaměstnanec se přihlašuje pomocí svého firemního e-mailu a hesla, které si zvolil. Každému zaměstnanci jsou přidělena určitá práva. Práva vždy přiděluje Administrátor e-shopu. Máme tři základní úrovně práv:

- **Administrátor**

Nejvyšší úroveň, může upravovat strukturu e-shopu, uživatele, objednávky i produkty.

- **Zaměstnanec**

Označení uživatele s právy, díky kterým může nahlížet na jednotlivé objednávky, přidávat i mazat produkty a měnit stavy jednotlivých objednávek. Stále ale nemá možnost spravovat ostatní uživatele.

- **Zaměstnanec s omezenými právy**

Výchozí typ uživatele, nemá možnost spravovat ostatní uživatele, objednávky ani zboží. Má práva pouze pro čtení.

## **4.3 Konceptuální návrh databáze**

V konceptuálním návrhu si identifikujeme základní entity, vysvětlíme si, co si pod nimi představit a uvedeme si i vazby mezi nimi. Nakonec si ukážeme, jak vypadá zjednodušený E-R diagram. Celý E-R diagram je pak v příloze.

### **4.3.1 Definice základních entit**

Zde se definujeme základní entity, které v databázi hrají roli.

#### **Zaměstnanec**

Zaměstnancem je každý, kdo ve firmě pracuje. Jsou zde tři typy zaměstnanců podle toho, jaká práva jim jsou přidělena. Je to administrátor, který má největší pravomoc a má přístup ke všem údajům, nastavením e-shopu a může i přidávat ostatní zaměstnance do administrace. Nižší úrovní pak je zaměstnanec, který může upravovat a vyřizovat objednávky. Nejnížší úroveň je pak zaměstnanec s omezenými právy, který má pouze práva pro čtení bez úprav, může tedy například nahlížet do objednávek, ale už nemůže měnit jejich stavy.

#### **Zákazníka**

Zákazníkem je každý, kdo e-shopu o sobě poskytl kontaktní informace. Pokud se zákazník registroval, při dalším nákupu už o sobě nemusí zadávat kontaktní informace, protože jsou už předvyplněny z databáze zákazníků. Nákup je pak komfortnější a rychlejší.

#### **Zboží**

Zbožím je vše, co nabízíme v našem e-shopu zákazníkům. Cílem návrhu tabulky zboží je dostatečná informovanost o zboží. Čím více informací e-shop o zboží poskytuje, tím větší je pravděpodobnost, že zákazník nakoupí. Vyšší informovanost totiž v zákazníkovi vyvolává větší pocit důvěrnosti či bezpečí.

#### **Objednávka**

Objednávka je proces mezi zákazníkem a prodejcem, kdy zákazník platí za dodání vybraného zboží prodejci. O objednávce potřebujeme znát informace o tom, komu má být

zboží doručeno a na jakou adresu, informace o vybraném zboží, způsobu doručení, typu platby a spoustu dalších. Všechny potřebné máme v návrhu v tabulce Objednávka.

## **Sklad**

Sklad je základem každého e-shopu i v případě, když zboží aktualizujeme pomocí XML feedu. Bez skladu bychom nemohli hlídat skladové zásoby a ani přiřazovat dostupnost zboží (např. Při 0 ks na skladě přiřadíme dostupnost „Vyprodáno“). V navrhované databázi budou „dva sklady“ („Vratky“ a sklad „Zelený“).

## **Doklad**

Dokladem se rozumí jakýkoliv dokument, který má strukturu podle české legislativy. Může jim být faktura či jiný doklad. Každý doklad tedy musí obsahovat určité informace, aby byl platný.

*„Na dokladu musí být uvedeno označení podnikatele obchodní firmou, popřípadě názvem nebo jménem a příjmením, a identifikačním číslem, datum prodeje zboží nebo poskytnutí služby, druh zboží nebo služby a cena, pokud zvláštní právní předpis nestanoví jinak“ (14).*

## **Faktura**

Na faktuře pak ještě musí navíc být pořadové číslo faktury a rok, datum vystavení a splatnosti faktury, číslo bankovního účtu, kód banky a způsob platby (15).

## **EET účtenka**

*„Povinné informace, které musí účtenka mít, jsou:*

- *fiskální identifikační kód,*
- *bezpečnostní kód poplatníka,*



- *daňové identifikační číslo podnikatele,*
- *označení provozovny a pokladního zařízení,*
- *datum a čas tržby,*
- *údaje, zda je tržba evidována v běžném, nebo zjednodušeném režimu“(16).*

#### 4.3.2 Vztahy mezi základními entitami

Jak jsme si již řekli, vztahy mezi entitami mohou být typu 1:1, 1:N, N:1 a N:M. Nyní si uvedeme vztahy, které jsou pro nás důležité.

##### **Zaměstnanec – Objednávka**

**Relace 1:N** -Zaměstnanec může mít na starost více objednávek, ale o jednu konkrétní objednávku se stará jen jeden zaměstnanec.

##### **Zákazník – Objednávka**

**Relace 1:N**-Zákazník může realizovat více objednávek, ovšem jedna konkrétní objednávka může být realizována pouze jedním zákazníkem.

##### **Zboží – Objednávka**

**Relace M:N** -Zboží může být součástí více objednávek, stejně tak jedna konkrétní objednávka může obsahovat více zboží. Budeme muset tedy provést dekompozici, o které si povíme později.

##### **Zboží – Sklad**

**Relace M:N** -Jedno konkrétní zboží se může nacházet na více skladech a stejně tak v jednom skladě najdeme více zboží.

### Zboží – Kategorie

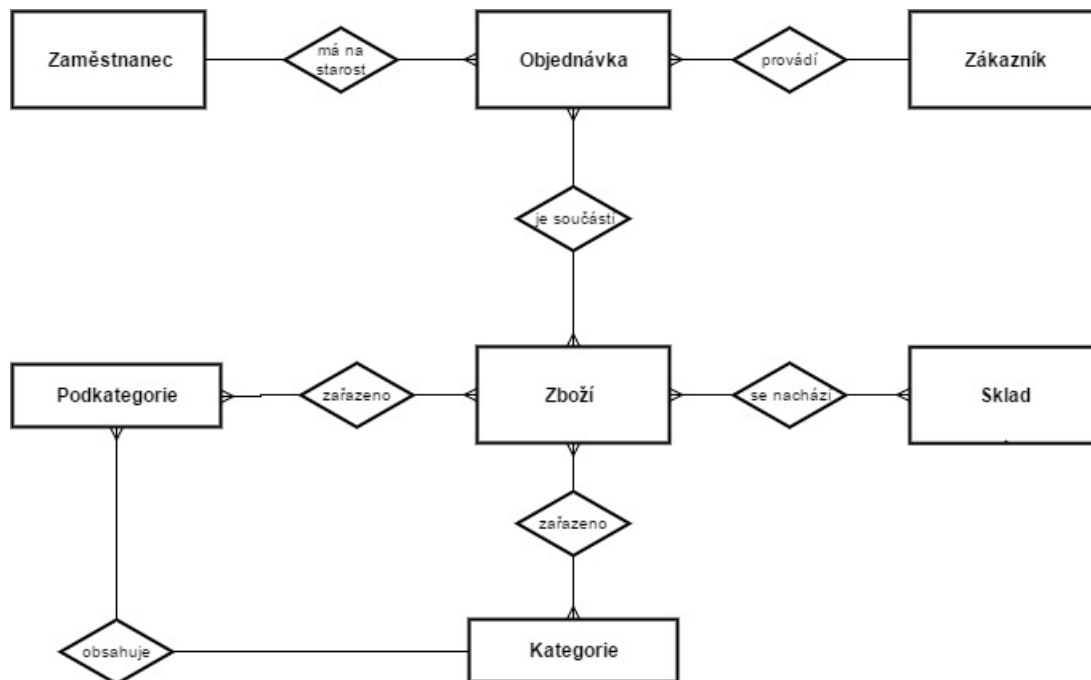
**Relace M:N** - Jedno konkrétní zboží může být zařazeno ve více kategoriích (např. kravata, boty Converse unisex mohou být jak v kategorii „pánské“ tak i „dámské“) a v jedné kategorii je zařazeno více zboží.

### Zboží – Podkategorie

**Relace M:N** - Stejně tak bude vypadat relace Zboží – Podkategorie, protože jedno konkrétní zboží může být zařazeno do více podkategorií a v jedné podkategorii je zařazeno více zboží.

### Kategorie – Podkategorie

**Relace 1:N** - Kategorie může obsahovat více podkategorií, ale jedna konkrétní podkategorie je obsažena jen v jedné kategorii.



Obr. 14: Základní E-R diagram před dekompozicí (vlastní zpracování)

## 4.4 Logický návrh databáze

V logickém návrhu si uvedeme postup řešení dekompozice vazeb M:N, projdeme si vytvořené entity (tabulky), nadefinujeme si atributy entit a jejich integritní omezení.

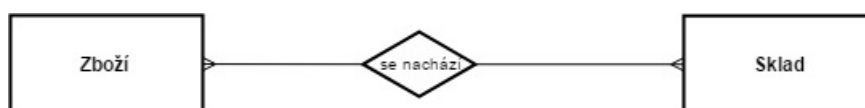
### 4.4.1 Dekompozice vazeb M:N

U některých vztahů jsme si stanovili vazbu M:N a přesně tuhle vazbu musíme dál rozložit. Dělá se to tak, že vytvoříme třetí tabulku, která bude obsahovat primární klíče z obou původních tabulek. V nově vytvořené tabulce budou ovšem tyhle primární klíče klíči cizími.

Pro lepší pochopení si vše ukážeme a detailně popíšeme na příkladu dekompozice relace M:N mezi zbožím a skladem.

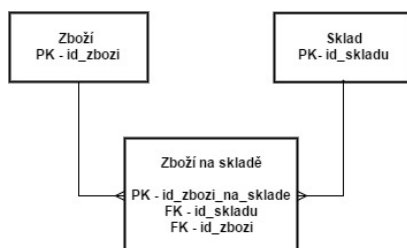
#### Původní relace

Takhle vypadá relace před dekompozicí



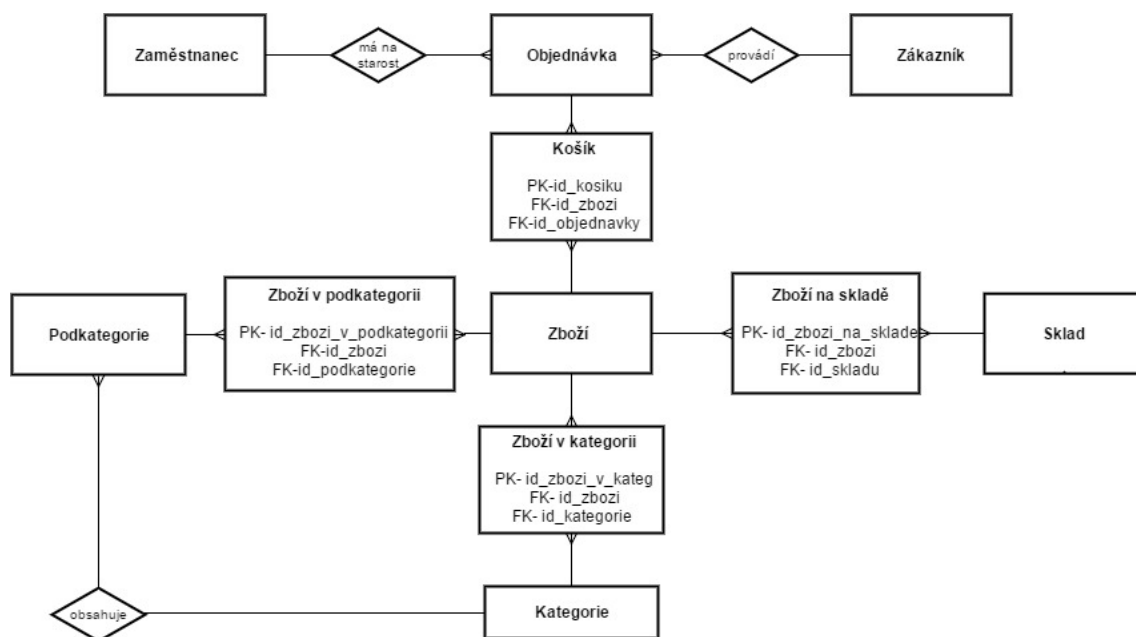
Obr. 15: Původní relace (vlastní zpracování)

## Dekompozice relace



Obr. 16: Dekompozice relace (vlastní zpracování)

Dekompozici jsme tedy provedli vytvořením třetí „mezi“ tabulky „Zboží na skladě“. Tato tabulka je, jak vidíme ve vztahu N:1 k oběma původním tabulkám. V tabulce „Zboží na skladě“ se nachází primární klíč tabulky „Zboží“ (id\_zbozi) a také primární klíč tabulky „Sklad“ (id\_skladu). Důležité je poznamenat, že oba primární klíče jsou v nově vytvořené tabulce klíči cizími.



Obr. 17: Základní E-R diagram po dekompozici (vlastní zpracování)

#### 4.4.2 Přehled všech vytvořených tabulek

Tab.4: DPH (vlastní zpracování)

| Integritní omezení | Atribut   | Datový typ   | Omezení/podmínky | Popis                        |
|--------------------|-----------|--------------|------------------|------------------------------|
| PK                 | id_dph    | INT          | not null         |                              |
|                    | sazba_dph | numeric(2,2) | not null         | sazba dph ve formě 0.21(21%) |

Tab. 5: STÁT (vlastní zpracování)

| Integritní omezení | Atribut  | Datový typ  | Omezení/podmínky | Popis                        |
|--------------------|----------|-------------|------------------|------------------------------|
| PK                 | id_statu | INT         | not null         |                              |
|                    | nazev    | varchar(20) | not null         | sazba dph ve formě 0.21(21%) |
|                    | zkratka  | varchar(3)  | not null         | zkratka státu (ČR, SVK...)   |

Tab. 6: JEDNOTKY (vlastní zpracování)

| Integritní omezení | Atribut     | Datový typ  | Omezení/podmínky | Popis                             |
|--------------------|-------------|-------------|------------------|-----------------------------------|
| PK                 | id_jednotky | INT         | not null         |                                   |
|                    | nazev       | varchar(20) | not null         | název jednotky (kilogram, kus...) |
|                    | zkratka     | varchar(4)  | not null         | zkratka jednotky (kg, ks...)      |

Tab. 7: OBRÁZKY (vlastní zpracování)

| Integritní omezení | Atribut    | Datový typ   | Omezení/podmínky | Popis                             |
|--------------------|------------|--------------|------------------|-----------------------------------|
| PK                 | id_obrazku | INT          | not null         |                                   |
|                    | obrazek    | img          | not null         | název jednotky (kilogram, kus...) |
|                    | alt_nazev  | varchar(100) | not null         | zkratka jednotky (kg, ks...)      |
|                    | rozliseni  | varchar(15)  |                  | rozlišení obrázku (1280x960...)   |

Tab. 8: STAV OBJEDNÁVKY (vlastní zpracování)

| Integritní omezení | Atribut            | Datový typ  | Omezení/podmínky | Popis   |
|--------------------|--------------------|-------------|------------------|---|
| PK                 | id_stav_objednavky | INT         | not null         |   |
|                    | nazev              | varchar(20) | not null         | stav objednávky (Čeká na platbu, Vyřizuje se, Dokončena, Zrušena) |

Tab. 9: STAV PLATBY (vlastní zpracování)

| Integritní omezení | Atribut        | Datový typ  | Omezení/podmínky | Popis                                |
|--------------------|----------------|-------------|------------------|--------------------------------------|
| PK                 | id_stav_platby | INT         | not null         |                                      |
|                    | nazev          | varchar(20) | not null         | stav platby (Zaplaceno, Nezaplaceno) |

Tab. 10: TYP PLATBY (vlastní zpracování)

| Integritní omezení | Atribut       | Datový typ   | Omezení/podmínky | Popis  |
|--------------------|---------------|--------------|------------------|--|
| PK                 | id_typ_platby | INT          | not null         |  |
|                    | nazev         | varchar(20)  | not null         | typ platby (Online kartou, dobírkou, hotově, převodem) |
|                    | cena          | numeric(6,2) | not null         | cena za platbu (u dobírky)                             |

Tab. 11: ZPŮSOB DORUČENÍ (vlastní zpracování)

| Integritní omezení | Atribut            | Datový typ   | Omezení/podmínky | Popis                                 |
|--------------------|--------------------|--------------|------------------|---------------------------------------|
| PK                 | id_zpusob_doruceni | INT          | not null         |                                       |
|                    | nazev              | varchar(60)  | not null         | dopravci (GLS, DPD, PPL, Česká pošta) |
|                    | doba_doruceni      | varchar(10)  | not null         | dodací doba                           |
|                    | cena               | numeric(6,2) | not null         | cena za doručení/dopravu              |

Tab. 12: BARVA (vlastní zpracování)

| Integritní omezení | Atribut  | Datový typ  | Omezení/podmínky | Popis                              |
|--------------------|----------|-------------|------------------|------------------------------------|
| PK                 | id_barvy | INT         | not null         |                                    |
|                    | nazev    | varchar(20) | not null         | barva produktu (modrá, červená...) |

Tab. 13: MATERIÁL (vlastní zpracování)

| Integritní omezení | Atribut      | Datový typ  | Omezení/podmínky | Popis   |
|--------------------|--------------|-------------|------------------|---|
| PK                 | id_materialu | INT         | not null         |   |
|                    | nazev        | varchar(20) | not null         | materiál produktu (kov a jiné, bavlna, vlna...) |

Tab. 14: VELIKOST (vlastní zpracování)

| Integritní omezení | Atribut      | Datový typ | Omezení/podmínky | Popis                        |
|--------------------|--------------|------------|------------------|------------------------------|
| PK                 | id_velikosti | INT        | not null         |                              |
|                    | hodnota      | varchar(4) | not null         | velikosti (45, 46, S, M,...) |

Tab. 15: ŠÍŘKA (vlastní zpracování)

| Integritní omezení | Atribut  | Datový typ  | Omezení/podmínky | Popis                   |
|--------------------|----------|-------------|------------------|-------------------------|
| PK                 | id_sirky | INT         | not null         |                         |
|                    | nazev    | varchar(20) | not null         | Rozměry (4 cm, 5 cm...) |

Tab. 16: GARANCE (vlastní zpracování)

| Integritní omezení | Atribut    | Datový typ  | Omezení/podmínky | Popis                          |
|--------------------|------------|-------------|------------------|--------------------------------|
| PK                 | id_garance | INT         | not null         |                                |
|                    | nazev      | varchar(10) | not null         | doba garance/záruky v měsících |

Tab. 17: VÝROBCE (vlastní zpracování)

| Integritní omezení | Atribut    | Datový typ  | Omezení/podmínky | Popis         |
|--------------------|------------|-------------|------------------|---------------|
| PK                 | id_vyrobce | INT         | not null         |               |
|                    | nazev      | varchar(30) | not null         | název výrobce |

Tab. 18: PRAVOMOC (vlastní zpracování)

| Integritní omezení | Atribut                   | Datový typ  | Omezení/podmínky | Popis  |
|--------------------|---------------------------|-------------|------------------|--|
| PK                 | id_pravomoc               | INT         | not null         |  |
|                    | nazev                     | varchar(20) | not null         | stav platby (Zaplaceno, Nezaplaceno)                           |
|                    | administrator             | BIT         |                  | pravomoc administrátora (má veškerá práva)(0/1)                |
|                    | zamestnanec               | BIT         |                  | pravomoc zaměstnance (má práva na čtení a úpravy) (0/1)        |
|                    | zamestnanec_omezena_prava | BIT         |                  | pravomoc zaměstnance s omezenými právy (práva pro čtení) (0/1) |
|                    | pravomoc_pridelena        | date        | not null         | datum přidělení pravomoci                                      |

Tab. 19: ZAMĚSTNANEC (vlastní zpracování)

| Integritní omezení | Atribut          | Datový typ  | Omezení/podmínky | Popis                        |
|--------------------|------------------|-------------|------------------|------------------------------|
| PK                 | id_zamestnanec   | INT         | not null         |                              |
|                    | jmeno            | varchar(25) | not null         | jméno zaměstnance            |
|                    | prijmeni         | varchar(35) | not null         | příjmení zaměstnance         |
|                    | email            | varchar(60) | not null         | email zaměstnance            |
|                    | heslo            | varchar(50) | not null         | heslo pro přihlášení         |
|                    | datum_registrace | date        | not null         | datum registrace zaměstnance |
| FK                 | id_pravomoc      | INT         | not null         |                              |

Tab.20: KATEGORIE (vlastní zpracování)

| Integritní omezení | Atribut      | Datový typ   | Omezení/podmínky | Popis                           |
|--------------------|--------------|--------------|------------------|---------------------------------|
| PK                 | id_kategorie | INT          | not null         |                                 |
|                    | nazev        | varchar(50)  | not null         | název kategorie                 |
|                    | url          | varchar(100) | not null         | url kategorie (/panske_kravaty) |
|                    | popis        | varchar(500) | not null         | popis kategorie                 |

Tab. 20: PODKATEGORIE (vlastní zpracování)

| Integritní omezení | Atribut         | Datový typ   | Omezení/podmínky | Popis                           |
|--------------------|-----------------|--------------|------------------|---------------------------------|
| PK                 | id_podkategorie | INT          | not null         |                                 |
|                    | nazev           | varchar(50)  | not null         | název kategorie                 |
|                    | url             | varchar(100) | not null         | url kategorie (/panske_kravaty) |
|                    | popis           | varchar(500) | not null         | popis kategorie                 |
| FK                 | id_kategorie    | INT          | not null         |                                 |

Tab.21: DORUČOVACÍ ADRESA (vlastní zpracování)

| Integritní omezení | Atribut         | Datový typ  | Omezení/podmínky | Popis                                  |
|--------------------|-----------------|-------------|------------------|--|
| PK                 | id_doruc_adresa | INT         | not null         |  |
|                    | město           | varchar(30) | not null         | název kategorie                        |
|                    | psc             | varchar(5)  | not null         | url adresa kategorie (/panske_kravaty) |
|                    | ulice           | varchar(50) | not null         | popis kategorie                        |
|                    | c_domu          | varchar(5)  | not null         | číslo domu                             |
| FK                 | id_statu        | INT         | not null         |  |

Tab. 22: FAKTURAČNÍ ADRESA (vlastní zpracování)

| Integritní omezení | Atribut        | Datový typ  | Omezení/podmínky | Popis                                  |
|--------------------|----------------|-------------|------------------|--|
| PK                 | id_fakt_adresa | INT         | not null         |  |
|                    | město          | varchar(30) | not null         | název kategorie                        |
|                    | psc            | varchar(5)  | not null         | url adresa kategorie (/panske_kravaty) |
|                    | ulice          | varchar(50) | not null         | popis kategorie                        |
|                    | c_domu         | varchar(5)  | not null         | číslo domu                             |
| FK                 | id_statu       | INT         | not null         |  |

Tab. 23: REGISTRACE (vlastní zpracování)

| Integritní omezení | Atribut          | Datový typ  | Omezení/podmínky | Popis                           |
|--------------------|------------------|-------------|------------------|---------------------------------|
| PK                 | id_registrace    | INT         | not null         |                                 |
|                    | email            | varchar(60) | not null         | název kategorie                 |
|                    | heslo            | varchar(30) | not null         | url kategorie (/panske_kravaty) |
|                    | datum_registrace | date        | not null         | popis kategorie                 |



Tab. 24: ZÁKAZNÍK (vlastní zpracování)

| Integritní omezení | Atribut         | Datový typ  | Omezení/podmínky | Popis  |
|--------------------|-----------------|-------------|------------------|--|
| PK                 | id_zakaznika    | INT         | not null         |  |
|                    | jmeno           | varchar(25) | not null         | stav platby (Zaplaceno, Nezaplaceno)                           |
|                    | prijmeni        | varchar(35) | not null         | pravomoc administrátora (má veškerá práva)(0/1)                |
|                    | email           | varchar(60) | not null         | pravomoc zaměstnance (má práva na čtení a úpravy) (0/1)        |
|                    | telefon         | varchar(13) |                  | pravomoc zaměstnance s omezenými právy (práva pro čtení) (0/1) |
|                    | na_firmu        | BIT         | not null         | datum přidělení pravomoci                                      |
|                    | ico             | varchar(13) |                  | IČO  |
|                    | dic             | varchar(14) |                  | DIČ  |
| FK                 | id_doruc_adresa | INT         | not null         |  |
| FK                 | id_fakt_adresa  | INT         | not null         |  |
| FK                 | id_registrace   | INT         |                  |  |

Tab. 25: REMARKETING (vlastní zpracování)

| Integritní omezení | Atribut                         | Datový typ | Omezení/podmínky | Popis  |
|--------------------|---------------------------------|------------|------------------|--|
| PK                 | id_remarketingu                 | INT        | not null         |  |
| FK                 | id_zakaznika                    | INT        | not null         |  |
|                    | jeden_az_tri_dny_na_vstivil_web | BIT        |                  | navštívení webu za posledních 1-3 dny              |
|                    | tri_az_sedm_dny_na_vstivil_web  | BIT        |                  | navštívení webu za posledních 3-7 dny              |
|                    | vlozili_do_kosiku30             | BIT        |                  | vložení produktu do košíku v posledních 30ti dnech |
|                    | newsletter                      | BIT        |                  | odebírání newsletteru                              |

Tab. 26: ZBOŽÍ (vlastní zpracování)

| Integritní omezení | Atribut       | Datový typ    | Omezení/podmínky | Popis                  |
|--------------------|---------------|---------------|------------------|------------------------|
| PK                 | id_zbozi      | INT           | not null         |                        |
|                    | nazev         | varchar(100)  | not null         | název zboží            |
|                    | cena_nakupni  | INT           | not null         | nákupní cena zboží     |
|                    | cena_prodejni | INT           | not null         | prodejní cena zboží    |
|                    | marze         | varchar(4)    |                  |                        |
|                    | ean           | varchar(13)   |                  |                        |
|                    | url_adresa    | varchar(100)  | not null         | url adresa zboží       |
| FK                 | id_vyrobce    | INT           | not null         |                        |
| FK                 | id_garance    | INT           | not null         |                        |
| FK                 | id_obrazku    | INT           |                  |                        |
| FK                 | id_dph        | INT           | not null         |                        |
| FK                 | id_barvy      | INT           | not null         |                        |
| FK                 | id_sirky      | INT           |                  |                        |
| FK                 | id_velikosti  | INT           |                  |                        |
| FK                 | id_materialu  | INT           | not null         |                        |
| FK                 | id_jednotky   | INT           | not null         |                        |
|                    | kratky_popis  | varchar (200) | not null         | krátký popis zboží     |
|                    | dlouhy_popis  | varchar (500) | not null         | dlouhý popis zboží     |
|                    | znacka_akce   | BIT           | not null         | štítek „akce“ u zboží  |
|                    | znacka_sleva  | BIT           | not null         | štítek „sleva“ u zboží |

Tab. 27: SKLAD (vlastní zpracování)

| Integritní omezení | Atribut      | Datový typ  | Omezení/podmínky | Popis |
|--------------------|--------------|-------------|------------------|-------|
| PK                 | id_skladu    | INT         | not null         |       |
|                    | nazev_skladu | varchar(30) | not null         |       |

Tab. 28: ZBOŽÍ NA SKLADĚ (vlastní zpracování)

| Integritní omezení | Atribut            | Datový typ  | Omezení/podmínky | Popis                                |
|--------------------|--------------------|-------------|------------------|--------------------------------------|
| PK                 | id_zbozi_na_sklade | INT         | not null         |                                      |
| FK                 | id_zbozi           | INT         | not null         |                                      |
| FK                 | id_skladu          | INT         | not null         |                                      |
| FK                 | id_jednotky        | INT         | not null         |                                      |
|                    | dostupnost         | varchar(15) | not null         | Dostupnost zboží (Skladem/Vyprodáno) |
|                    | mnozstvi           | varchar(4)  | not null         | Počet ks na skladě                   |
|                    | datum_aktualizace  | date        | not null         | Datum aktualizace zboží              |

Tab. 29: OBJEDNÁVKA (vlastní zpracování)

| Integritní omezení | Atribut            | Datový typ | Omezení/podmínky | Popis                                |
|--------------------|--------------------|------------|------------------|--------------------------------------|
| PK                 | id_objednavky      | INT        | not null         |                                      |
| FK                 | id_zbozi           | INT        | not null         |                                      |
|                    | datum_objednavky   | date       | not null         | Datum, kdy byla objednávka vytvořena |
|                    | update_objednavky  | date       | not null         | Datum aktualizace objednávky         |
| FK                 | id_stav_platby     | INT        | not null         |                                      |
| FK                 | id_typ_platby      | INT        | not null         |                                      |
| FK                 | id_stav_objednavky | INT        | not null         |                                      |
| FK                 | id_zpusob_doruceni | INT        | not null         |                                      |
| FK                 | id_zakaznika       | INT        | not null         |                                      |
| FK                 | id_zamestnance     | INT        | not null         |                                      |

Tab.30: KOŠÍK (vlastní zpracování)

| Integritní omezení | Atribut            | Datový typ | Omezení/podmínky | Popis             |
|--------------------|--------------------|------------|------------------|-------------------|
| PK                 | id_kosiku          | INT        | not null         |                   |
|                    | mnozstvi           | TINYINT    | not null         | Počet ks v košíku |
| FK                 | id_objednavky      | INT        | not null         |                   |
| FK                 | id_typ_platby      | INT        | not null         |                   |
| FK                 | id_zpusob_doruceni | INT        | not null         |                   |
| FK                 | id_zbozi           | INT        | not null         |                   |

Tab. 31: ZBOŽÍ V KATEGORII (vlastní zpracování)

| Integritní omezení | Atribut              | Datový typ | Omezení/podmínky | Popis |
|--------------------|----------------------|------------|------------------|-------|
| PK                 | id_zbozi_v_kategorii | INT        | not null         |       |
| FK                 | id_kategorie         | INT        | not null         |       |
| FK                 | id_zbozi             | INT        | not null         |       |

Tab. 32: ZBOŽÍ V PODKATEGORII (vlastní zpracování)

| Integritní omezení | Atribut                 | Datový typ | Omezení/podmínky | Popis |
|--------------------|-------------------------|------------|------------------|-------|
| PK                 | id_zbozi_v_podkategorii | INT        | not null         |       |
| FK                 | id_kategorie            | INT        | not null         |       |
| FK                 | id_zbozi                | INT        | not null         |       |

Tab. 33: REKLAMACE (vlastní zpracování)

| Integritní omezení | Atribut           | Datový typ   | Omezení/podmínky | Popis                   |
|--------------------|-------------------|--------------|------------------|-------------------------|
| PK                 | id_reklamace      | INT          | not null         |                         |
| FK                 | id_objednavky     | INT          | not null         |                         |
| FK                 | id_zbozi          | INT          | not null         |                         |
| FK                 | id_zakaznika      | INT          | not null         |                         |
|                    | duvod_reklamace   | varchar(500) | not null         | Důvod reklamace zboží   |
|                    | reklamovano_datum | date         | not null         | Datum přijetí reklamace |

Tab. 34: POKLADNÍ ZAŘÍZENÍ (vlastní zpracování)

| Integritní omezení | Atribut        | Datový typ  | Omezení/podmínky | Popis                |
|--------------------|----------------|-------------|------------------|----------------------|
| PK                 | id_pokladny    | INT         | not null         |                      |
|                    | nazev_pokladny | varchar(50) | not null         | Pojmenování pokladny |

Tab. 35: FAKTURA (vlastní zpracování)

| Integritní omezení | Atribut                | Datový typ  | Omezení/podmínky | Popis  |
|--------------------|------------------------|-------------|------------------|--|
| PK                 | cislo_faktury          | BIGINT      | not null         |  |
|                    | ico                    | varchar(13) | not null         | IČO  |
|                    | dic                    | varchar(14) | not null         | DIČ  |
|                    | jmeno_podnikatele      | Varchar(40) | not null         | Jméno podnikatele na faktuře (informace o prodejci)  |
|                    | platce_dph             | BIT         | not null         | Informace, zda je plátcem DPH (informace o prodejci) |
|                    | c_bank_uctu            | varchar(20) | not null         | číslo bank.úctu podnikatele (informace o prodejci)   |
|                    | kod_banky              | varchar(4)  | not null         | kód banky  |
|                    | datum_vystaveni        | date        | not null         | datum vystavení faktury                              |
|                    | datum_splatnosti       | date        | not null         | datum splatnosti faktury                             |
|                    | cena_za_zbozi_a_sluzby | varchar(10) | not null         | celková cena na faktuře k zaplacení                  |
| FK                 | id_typ_platby          | INT         | not null         |  |

Tab. 36: EET účtenka (vlastní zpracování)

| Integrovní omezení | Atribut                     | Datový typ  | Omezení/podmínky | Popis                                   |
|--------------------|-----------------------------|-------------|------------------|---|
| PK                 | id_uctenky                  | INT         | not null         |   |
|                    | oznaceni_provozovny         | varchar(10) | not null         | Označení provozovny                     |
|                    | datum_trzby                 | date        | not null         | Datum kdy byla tržba přijata            |
|                    | danove_id_podnikatele       | varchar(10) | not null         | Daňové identifikační číslo podnikatele  |
|                    | bezpecnostni_kod_poplatnika | varchar(10) |                  | Bezpečnostní kód poplatníka (kvůli EET) |
|                    | fiskalni_identifikacni_kod  | varchar(10) |                  | Fiskální identifikační kód (kvůli EET)  |
| FK                 | id_pokladny                 | INT         | not null         |   |
| FK                 | id_typ_platby               | INT         | not null         |   |
|                    | cena_za_zbozi_a_sluzby      | varchar(10) |                  | celková cena na faktuře k zaplacení     |

Tab. 37: DOKLAD (vlastní zpracování)

| Integrovní omezení | Atribut             | Datový typ   | Omezení/podmínky | Popis  |
|--------------------|---------------------|--------------|------------------|--|
| PK                 | id_dokladu          | INT          | not null         |  |
|                    | id_objednavky       | INT          | not null         |  |
|                    | id_pokladny         | INT          | not null         |  |
|                    | id_zakaznika        | INT          | not null         |  |
|                    | id_zbozi            | INT          | not null         |  |
|                    | datum_cas_transakce | date         | not null         | Datum a čas transakce                                  |
|                    | udaje_o_trzbe       | varchar(100) |                  | Údaje za co byla tržba přijata (např. Prodej oblečení) |
| FK                 | cislo_faktury       | BIGINT       | not null         | Číslo faktury (např.2017000001)                        |
| FK                 | id_uctenky          | INT          |                  |  |

## Úplný E-R diagram

Úplný E-R naleznete v příloze.

## 4.5 Fyzický návrh databáze

Ve fyzickém návrhu se zaměříme na praktickou část, tedy na naprogramování databáze na základě předešlých informací a znalostí. Uvedeme si ale pouze části zdrojového kódu. Zbytek kódu je uveden v příloze.

#### 4.5.1 Volba rozhraní pro naprogramování databáze

Volit můžeme z nástrojů, které jsou zadarmo nebo zpoplatněné. Dá se říct, že výběr bychom měli přizpůsobit podle toho, co vše potřebujeme, aby nástroj uměl. Pokud naše požadavky splňuje nástroj, který je zadarmo, tak není důvod, proč za nástroj platit.

Já jsem si pro fyzický návrh databáze zvolil SQL Server 2014 Management Studio. Zvolil jsem si ho z důvodu pro mě již známého prostředí. Dříve jsem totiž již ve zmíněném nástroji pracoval a připadá mi uživatelsky příjemný. Abych byl přesný, pracoval jsem se starší verzí (2012). Já si ale řekl, že využiji novější verzi, když je dostupná.

#### 4.5.2 Triggery

Trigger funguje nad zvolenou tabulkou. Je to spoušť, díky které můžeme sledovat změny v databázi. Trigger se spouští příkazem (Insert, Delete, Update).

##### Oznámení nového uživatele

Trigger nás upozorní na vložení nového zaměstnance do databáze. Takový trigger může sloužit například Adminovi e-shopu.

##### --Vytvoření spouště

```
create trigger novy_zamestnanec ON zamestnanec
after insert
as
begin
PRINT 'Nový uživatel byl vložen do databáze'
end
go
```

##### --Smazání spouště

```
drop trigger nova_objednavka
```

## Upozornění na novou objednávku

Trigger nás upozorní na vznik nové objednávky

### --Vytvoření spouště

```
create trigger nova_objednavka
on objednavka
after insert
as
begin
print 'Vytvořena nová objednávka'
end
```

### --Smazání spouště

```
drop trigger nova_objednavka
```

## 4.5.3 Pohledy

Pohled si můžeme představit jako nějaké virtuální tabulky, které nám přináší určité výhody, jako je třeba zpřehlednění práce a ochrana dat (17).

### Přehled všech objednávek

Pohled, který nám zobrazí všechny objednávky. Řekne nám stav objednávky, ID objednávky, jméno a příjmení zákazníka, jaké zboží si objednal, cenu objednaného zboží, způsob doručení, doklady přiřazené k objednávce a datum aktualizace objednávky.

### --Vytvoření pohledu

```
create view prehled_vsech_objednavek
as
select stav_objednavky.nazev as 'Stav objednávky', objednavka.id_objednavky
as 'ID Objednávky', zakaznik.jmeno+' '+zakaznik.prijmeni as 'Zakazník', zbo-
zi.nazev as 'Zboží',
zbozi.cena_prodejni as 'Částka', zpusob_doruceni.nazev as 'Způsob doručení',
doklad.id_dokladu as 'ID dokladu',faktura.cislo_faktury as 'Faktura č.', para-
```

```
gon.id_paragonu as 'č. paragonu', objednavka.update_objednavky as 'Datum aktualizace objednávky'
```

```
from stav_objednavky, objednavka, zakaznik, zboží, zpusob_doruceni, kosik, doklad, faktura, paragon
```

```
where objednavka.id_stav_objednavky = stav_objednavky.id_stav_objednavky
and objednavka.id_zpusob_doruceni = zpusob_doruceni.id_zpusob_doruceni
and objednavka.id_zakaznika = zakaznik.id_zakaznika
and kosik.id_zbozi = zboží.id_zbozi
and kosik.id_objednavky = objednavka.id_objednavky
and faktura.cislo_faktury = doklad.cislo_faktury
and doklad.id_objednavky = objednavka.id_objednavky
and doklad.id_paragonu = paragon.id_paragonu
group by objednavka.id_objednavky, stav_objednavky.nazev, zakaznik.jmeno,
zakaznik.prijmeni, zboží.nazev, zboží.cena_prodejni, zpusob_doruceni.nazev, doklad.id_dokladu, paragon.id_paragonu,
faktura.cislo_faktury, objednavka.update_objednavky
go
```

#### **--Zobrazení pohledu**

```
select * from prehled_vsech_objednavek
```

### **Přehled zaměstnanců a objednávek, které mají na starost**

#### **--Vytvoření pohledu**

```
create view zamestnanec_obj
as
select zamestnanec.jmeno+' '+zamestnanec.prijmeni as 'Zaměstnanec', pravomoc.nazev as 'Pravomoc',
kosik.id_objednavky as 'Číslo objednávky na starost', zboží.nazev as 'Zboží', velikost.hodnota as 'Velikost', barva.nazev, sirka.nazev as 'Šířka',
material.nazev as 'Materiál', kosik.mnozstvi as 'Množství',
stav_objednavky.nazev as 'Stav objednávky'
from zamestnanec, pravomoc, objednavka, kosik, zboží, material, velikost, sirka,
barva, stav_objednavky
where zamestnanec.id_pravomoc = pravomoc.id_pravomoc
and objednavka.id_zamestnance = zamestnanec.id_zamestnance
and kosik.id_objednavky = objednavka.id_objednavky
and kosik.id_zbozi = zboží.id_zbozi
```

```

and zboží.id_barvy = barva.id_barvy
and zboží.id_materialu = material.id_materialu
and zboží.id_sirky = sirka.id_sirky
and zboží.id_velikosti = velikost.id_velikosti
and objednávka.id_stav_objednavky = stav_objednavky.id_stav_objednavky
go

```

#### **--Zobrazení pohledu**

```

select * from zamestnanec_obj

```

### **4.5.4 Procedury**

#### **Vyhledání zboží**

Asi nejdůležitější procedura, díky které můžeme vyhledat zboží. Vyhledání zboží je základní funkcí, kterou musí každý e-shop mít. Bez ní by totiž zákazník, který nezná strukturu e-shopu, nenajde to, co hledá. Výsledky vyhledávání jsou seřazeny od nejnižší ceny.

#### **--Vytvoření procedury**

```

create procedure hledat_zbozi
(@nazev nvarchar(100),
@ean char(13))
as
select zboží.ean as 'EAN', zboží.nazev as 'Název', zboží.kratky_popis as 'Krátký popis zboží', zboží.cena_prodejni as 'Cena'
from zboží
where zboží.nazev like '%'+@nazev+'%' or @ean=ean
order by zboží.cena_prodejni
go

```

#### **-- Spuštění procedury**

```

exec hledat_zbozi 'košile',null

```

#### **--Smazání procedury**

```

drop procedure hledat_zbozi

```



## Vyhledávání zákazníka podle jména nebo příjmení

Procedura nám po zadání části jména nebo příjmení vyhledá zaměstnance podle shody.

### --Vytvoření procedury

```
create procedure hledat_zak
@jmeno_prijmeni nvarchar(35)
as
select  zakaznik.id_zakaznika as 'ID Zákazníka', zakaznik.jmeno+'
'+zakaznik.prijmeni as 'Zákazník', zakaznik.email as 'E-mail', zakaznik.telefon as
'Telefon',dorucovaci_adresa.mesto+', '+dorucovaci_adresa.psc+',
'+dorucovaci_adresa.ulice+' '+dorucovaci_adresa.c_domu as 'Adresa'
from zakaznik, dorucovaci_adresa
where zakaznik.jmeno like '%'+@jmeno_prijmeni+'%' or zakaznik.prijmeni like
'%'+@jmeno_prijmeni+'%'
and zakaznik.id_doruc_adresa = dorucovaci_adresa.id_doruc_adresa
group by zakaznik.id_zakaznika, zakaznik.jmeno, zakaznik.prijmeni, zakaz-
nik.email, zakaznik.telefon, dorucovaci_adresa.mesto, dorucovaci_adresa.psc,
dorucovaci_adresa.ulice, dorucovaci_adresa.c_domu
order by zakaznik.id_zakaznika
go
```

### -- Spuštění procedury

```
exec hledat_zak 'nov'
```

### --Smazání procedury

```
drop procedure hledat_zak
```

## 4.5.5 Další postup po vlastním návrhu

Dalším krokem po tvorbě databáze je výběr řešení e-shopu. Obecně jsou možné asi tyto typy řešení:

- Naprogramovat si zbytek sám

Toto řešení je vhodné pro uživatele, který má znalosti s programováním e-shopů. Může si pak e-shop přizpůsobit přímo podle sebe (to platí i pro průběžné úpravy při tvorbě). Odpadají tu náklady na konzultace s programátorem či firmou i obtíže s domluvou během tvorby. E-shop nás pak zjednodušeně řečeno, stojí náš vlastní čas. E-shop je pak i naším majetkem a máme na něj výhradní práva.

- **Nechat si zbytek naprogramovat odborníkem**

Toto řešení je vhodné pro uživatele bez znalostí programování, kteří ovšem mají peníze, které můžou do tvorby e-shopu investovat. Víme, že e-shop bude na 99% funkční. Tvorba e-shopu je podložena smlouvou a chrání tak dodavatele služby, tak i zákazníka.

- **Vybrat placené komerční řešení**

Výhodou tohoto řešení je, že nám odpadají vysoké náklady na tvorbu. Řešení je vhodné pro začínající e-shopy. Zavazujeme se ale k pravidelné platbě ještě před spuštěním e-shopu, kdy ještě nevíme, zda bude e-shop vydělávat. Možnost zásadnějších úprav je zde často omezena. Nepotřebujeme ani výrazné znalosti programování. Příkladem takového řešení může být již zmiňovaný Shoptet, Magento, OpenCart nebo třeba oXyShop.

- **Vybrat neplacené komerční řešení**

Takové řešení má stejné výhody a nevýhody jako předchozí řešení s jediným rozdílem, že se nezavazujeme k pravidelné platbě. Toto řešení je tedy optimální pro začínající e-shopy. Mínusem tu zas bývá nízká až nulová podpora ze strany tvůrce řešení. Příkladem je třeba Prestashop.

#### 4.5.6 Zhodnocení a přínosy vlastního řešení

Cílem bakalářské práce bylo navrhnout SQL databázi pro elektronický obchod v SQL Server Management Studio 2014. Jsem názoru, že cíl byl splněn a navržená databáze může být dobrým základem pro nový e-shop. Samozřejmostí je, že pokud se má navržená databáze denně využívat, je třeba ještě spousta práce. Budeme potřebovat zkušeného odborníka, který ji aplikuje v praxi.

Je těžké zhodnotit ekonomický přínos databáze, když navržená databáze ještě není využívána. Ovšem můžeme odhadnout, jaké přínosy by mohla potenciálně přinést.

Databáze byla navržena tak, aby podporovala procesy jako je automatická aktualizaci a import zboží, zrychlila jednotlivé činnosti a navýšila efektivitu. V databázi není možné uložit záznamy bez vyplnění požadovaných informací.

Byla také rozšířena o remarketingovou část, která usnadní zaměstnancům čas při práci se zákazníky jako je segmentace a analýza.

Dále je tu pak rozšíření o vratky, které ušetří e-shopu tisíce korun ročně, a to díky lepšímu povědomí a informacích o dostupnosti zboží. Nebude už docházet k případu, kdy se zbytečně objednává nové zboží, přestože je zboží dostupné jako vrátka.

Z ekonomického hlediska by databáze mohla ušetřit desítky tisíc korun ročně. Úspora by měla být vidět především v remarketingové části, kde je potenciál ušetřit za práci zaměstnanců, kteří pracovali na databázi zákazníků. Ta by jim jinak zabrala desítky až stovky hodin ročně. Dále by měla několikanásobně zefektivnit jak přípravu, tak i provedení remarketingu, zvýšit počet objednávek a tím i zvýšit příjmy při nižších investicích (díky zvýšení relevantnosti cílení na zákazníky). Vratky by pak mohli dělat rozdíl v tisících až desetitisících ročně.

## ZÁVĚR

Cílem této práce bylo navrhnout SQL databázi pro elektronický obchod. Nejprve jsem si však identifikoval dílčí cíle, kterých je třeba dosáhnout.

Práce má tři hlavní kapitoly. Jsou to teoretická východiska potřebná pro pochopení, co to databáze jsou a jak s nimi pracovat.

Druhá kapitola je analýza současného stavu, kde jsem popsal dnešní fungování e-shopu, zanalyzoval jeho důležité procesy a vytkl nedostatky, které dané procesy mají. Procesy jsem se snažil popsat srozumitelně pomocí vývojových diagramů, aby je pochopil i lajk, který s problematikou databází nemá zkušenosti.

V třetí části, návrhu vlastního řešení, jsem nastínil, jak by mnou navržená databáze měla procesně fungovat (opět pomocí vývojových diagramů) a vysvětlil jsem změny a výhody oproti původnímu řešení. Dále jsem i stanovil požadavky na databázi jako je aktuálnost informací, určitý stupeň zabezpečení ve formě přístupových práv, jednoduchost databáze a další. Nechybí i identifikace základních entit a jejich popis. V návrhu vlastního řešení je zahrnut i logický a fyzická návrh. V logickém návrhu se zabývám problematikou dekompozice vazeb M:N, kterou nastiňuji na konkrétním příkladu. Součástí jsou dále i všechny vytvořené tabulky s popisem atributů a datových typů. Ve fyzickém návrhu zase řeším přímo části zdrojového kódu, který jsem naprogramoval. Nechybí zde ani ukázka a vysvětlení triggerů, procedur a pohledů.

Výsledkem je funkční databáze, která splňuje stanovené požadavky. Databáze má potenciál ušetřit e-shopu tisíce korun ročně, a naopak i zefektivnit některé procesy. Je nutno podotknout, že s databází v tomhle stavu, by uživatelé nedokázali pracovat. Je proto potřeba databázi napojit na nějakou aplikaci, která bude fungovat nad databází tak, aby byla přívětivá pro uživatele a oni s ní dokázali pracovat.

## SEZNAM POUŽITÝCH ZDROJŮ

- [1] LACKO, Luboslav. *Mistrovství v SQL Server 2012*. Brno: Computer Press, 2013, 640 s. : il. ISBN 978-80-251-3773-4.
- [2] KROENKE, David, David J AUER a Jakub GONER. *Databáze*. Brno: Computer Press, 2015, 496 s.: il., grafy, tab. ISBN 978-80-251-4352-0.
- [3] CEJPEK, Jiří. *Informace, komunikace a myšlení: úvod do informační vědy*. 2. přeprac. vyd. Praha: Karolinum, 2005, 233 s. ISBN 80-246-1037-X.
- [4] KOCH, M.; NEUWIRTH, B. *Datové a funkční modelování*. Brno: Cerm, 2010. ISBN: 978-80-214-4125- 5
- [5] STEPHENS, Ryan K, Ronald R PLEW, Arie JONES a Lukáš KREJČÍ. *Naučte se SQL za 28 dní*. Brno: Computer Press, 2010, 728 s. : il. ISBN 978-80-251-2700-
- [6] Hierarchie Data → Informace → Znalost – Wikisofia. [online]. Copyright © 2013 ISSN [cit. 21.05.2017]. Dostupné z: [https://wikisofia.cz/wiki/Hierarchie\\_Data\\_%E2%86%92\\_Informace\\_%E2%86%92\\_Znalost](https://wikisofia.cz/wiki/Hierarchie_Data_%E2%86%92_Informace_%E2%86%92_Znalost)
- [7] Co je to software? *IT Slovník* [online]. Copyright © 2017 [cit. 21.05.2017]. Dostupné z: [https://itslovník.cz/pojem/software/?utm\\_source=cp&utm\\_medium=link&utm\\_campaign=cp](https://itslovník.cz/pojem/software/?utm_source=cp&utm_medium=link&utm_campaign=cp)
- [8] Rozdíl mezi software a hardware – Rozdíly.cz. *Rozdíly.cz* [online]. Dostupné z: [http://www.rozdily.cz/Rozd%C3%ADl\\_mezi\\_software\\_a\\_hardware](http://www.rozdily.cz/Rozd%C3%ADl_mezi_software_a_hardware)
- [9] Lehký úvod – teorie databází. *Největší český web zaměřený na .NET framework* [online]. Copyright © 2017 [cit. 21.05.2017]. Dostupné z: <http://www.dotnetportal.cz/clanek/60/Lehky-uvod-teorie-databazi>

- [10] Databáze a jazyk SQL | Interval.cz. *Interval.cz | Svět Internetu, Technologií a Bezpečnosti* [online]. Copyright © [cit. 21.05.2017]. Dostupné z: <http://www.interval.cz/clanky/databaze-a-jazyk-sql/>
- [11] Objektové databáze - Linux E X P R E S. *Linux E X P R E S* [online]. Copyright © 2017 CCB, spol. s r. o., všechna práva vyhrazena. [cit. 21.05.2017]. Dostupné z: <https://www.linuxexpres.cz/business/objektove-databaze>
- [12] Databáze. Databáze [online]. Copyright © 2010 Misha [cit. 21.05.2017]. Dostupné z: <http://www.databaze.chytrak.cz/modely.htm>
- [13] Česká eshopová platforma Shoptet tento rok očekává růst obrátu o 82 % na 50 milionů Kč | CzechCrunch – nejčtenější magazín o startupech a technologiích. *CzechCrunch – nejčtenější magazín o startupech a technologiích* [online]. Copyright © Copyright 2014 [cit. 21.05.2017]. Dostupné z: <http://www.czechcrunch.cz/2016/07/ceska-eshopova-platforma-shoptet-tento-rok-ocekava-rust-obratu-o-82-na-50-milionu-kc/>
- [14] Zákon č. 455/1991 Sb., o živnostenském podnikání (živnostenský zákon)
- [15] Jak má vypadat faktura a co musí obsahovat | JakTak.cz. *JakTak.cz | Rady, návody a postupy* [online]. Copyright © 2017 www.jaktak.cz [cit. 21.05.2017]. Dostupné z: <http://www.jaktak.cz/jak-ma-vypadat-faktura-co-musi-obsahovat.html>
- [16] Co všechno musí být na účtence, abyste kvůli #EET nemuseli vydávat dva doklady? - Podnikatel.cz. *Podnikatel.cz - průvodce vaším podnikáním* [online]. Copyright © 2007 [cit. 21.05.2017]. Dostupné z: <http://www.podnikatel.cz/clanky/co-vsechno-musi-byt-na-uctence-abyste-kvuli-eet-nemuseli-vydat-dva-doklady/>

- [17] SQL – spojování tabulek a tvorba pohledů | Interval.cz. *Interval.cz | Svět Internetu, Technologií a Bezpečnosti* [online]. Copyright © [cit. 21.05.2017]. Dostupné z: <https://www.interval.cz/clanky/sql-spojovani-tabulek-a-tvorba-pohledu/>
- [18] Dnešní témata Technologie pořádání informací a znalostí v 21. století. *SlidePlayer* [online]. Copyright © 2017 SlidePlayer.cz Inc. [cit. 20.05.2017]. Dostupné z: <http://slideplayer.cz/slide/3280471/>
- [19] Coach's Guide to Dropshipping - Store Coach. *How to Start an Online Store << StoreCoach.com* [online]. Copyright © 2017 [cit. 20.05.2017]. Dostupné z: <https://storecoach.com/coachs-guide-to-dropshipping>

## SEZNAM OBRÁZKŮ

|  |    |
|--|----|
| OBR. 1: VZTAH DAT, INFORMACÍ A ZNALOSTÍ .....        | 13 |
| OBR. 2: HIERARCHICKÁ DATABÁZE .....                  | 19 |
| OBR. 3: SÍŤOVÁ DATABÁZE.....                         | 20 |
| OBR. 4: VAZBY MEZI ENTITAMI.....                     | 22 |
| OBR. 5: LOGO E-SHOPU .....                           | 27 |
| OBR. 6: ORGANIZAČNÍ STRUKTURA .....                  | 28 |
| OBR. 7: THE DROPSHIP MODEL .....                     | 31 |
| OBR. 8: REGISTRACE ZÁKAZNÍKA .....                   | 33 |
| OBR. 9: VYTVOŘENÍ OBJEDNÁVKY .....                   | 35 |
| OBR. 10: AKTUALIZACE DAT .....                       | 37 |
| OBR. 11: REGISTRACE ZÁKAZNÍKA - NOVĚ .....           | 39 |
| OBR. 12: VYTVOŘENÍ OBJEDNÁVKY - NOVĚ.....            | 41 |
| OBR. 13: AKTUALIZACE DAT - NOVĚ.....                 | 43 |
| OBR. 14: ZÁKLADNÍ E-R DIAGRAM PŘED DEKOMPOZICÍ ..... | 50 |
| OBR. 15: PŮVODNÍ RELACE.....                         | 51 |
| OBR. 16: DEKOMPOZICE RELACE .....                    | 52 |
| OBR. 17: ZÁKLADNÍ E-R DIAGRAM PO DEKOMPOZICI .....   | 52 |



## SEZNAM TABULEK

|                                  |    |
|----------------------------------|----|
| Tab. 1: 1. NORMÁLNÍ FORMA .....  | 18 |
| Tab. 2: JEDNOTKY .....           | 20 |
| Tab. 3: ZKRATKY .....            | 21 |
| Tab. 4: DPH .....                | 53 |
| Tab. 5: STÁT .....               | 53 |
| Tab. 6: JEDNOTKY .....           | 53 |
| Tab. 7: OBRÁZKY .....            | 53 |
| Tab. 8: STAV OBJEDNÁVKY .....    | 53 |
| Tab. 9: STAV PLATBY .....        | 53 |
| Tab. 10: TYP PLATBY .....        | 54 |
| Tab. 11: ZPŮSOB DORUČENÍ .....   | 54 |
| Tab. 12: BARVA .....             | 54 |
| Tab. 13: MATERIÁL .....          | 54 |
| Tab. 14: VELIKOST .....          | 54 |
| Tab. 15: ŠÍŘKA .....             | 54 |
| Tab. 16: GARANCE .....           | 55 |
| Tab. 17: VÝROBCE .....           | 55 |
| Tab. 18: PRAVOMOC .....          | 55 |
| Tab. 19: ZAMĚSTNANEC .....       | 55 |
| Tab. 20: KATEGORIE .....         | 55 |
| Tab. 20: PODKATEGORIE .....      | 56 |
| Tab. 21: DORUČOVACÍ ADRESA ..... | 56 |
| Tab. 22: FAKTURAČNÍ ADRESA ..... | 56 |
| Tab. 23: REGISTRACE .....        | 56 |
| Tab. 24: ZÁKAZNÍK .....          | 57 |
| Tab. 25: REMARKETING .....       | 57 |
| Tab. 26: ZBOŽÍ .....             | 57 |
| Tab. 27: SKLAD .....             | 58 |
| Tab. 28: ZBOŽÍ NA SKLADĚ .....   | 58 |
| Tab. 29: OBJEDNÁVKA .....        | 58 |

|                                    |    |
|------------------------------------|----|
| Tab. 30: KOŠÍK.....                | 58 |
| Tab. 31: ZBOŽÍ V KATEGORII.....    | 58 |
| Tab. 32: ZBOŽÍ V PODKATEGORII..... | 59 |
| Tab. 33: REKLAMACE .....           | 59 |
| Tab. 34: POKLADNÍ ZAŘÍZENÍ .....   | 59 |
| Tab. 35: FAKTURA .....             | 59 |
| Tab. 36: EET účtenka .....         | 60 |
| Tab. 37: DOKLAD .....              | 60 |

## SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

|             |  |
|-------------|--|
| E-R DIAGRAM | entity-relationship diagram              |
| E-SHOP      | elektronický obchod                      |
| SŘBD        | system řízení báze dat                   |
| PK          | primární klíč (primary key)              |
| FK          | cizí klíč (foreign key)                  |
| OID         | object identifier                        |
| EET         | elektornická evidence tržeb              |
| SQL         | Structured Query Language                |
| ANSI        | American National Standards Organization |
| ISO         | International Standards Organization     |

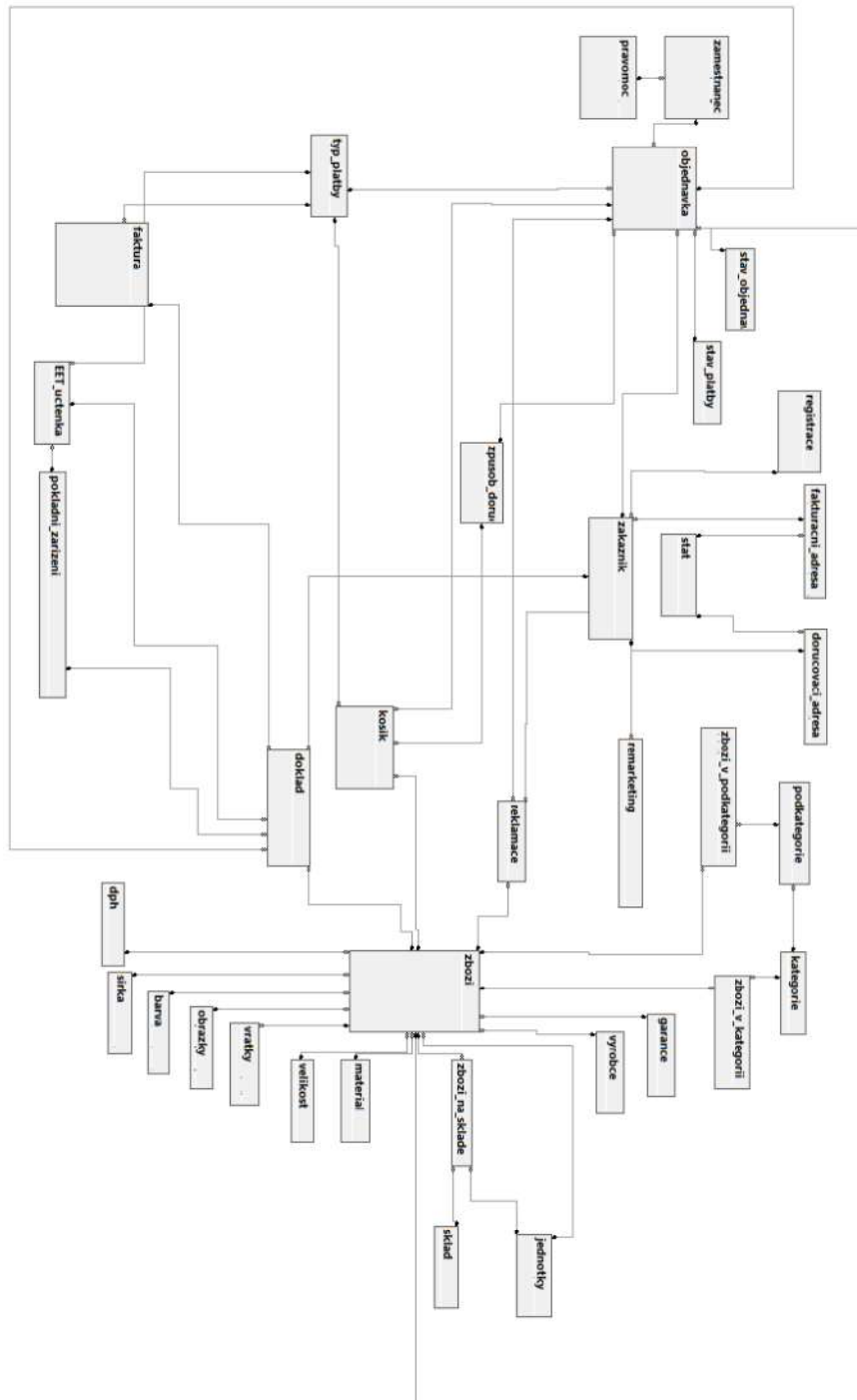
## **SEZNAM PŘÍLOH**

Příloha 1 - Úplný E-R diagram

Příloha 2 - Zdrojový kód

## PŘÍLOHY

## Příloha 1 - Úplný E-R diagram



## Příloha 2 - Zdrojový kód

```
create database eshop
go
use eshop
go
```

```
create table dph
(
id_dph INT identity (1, 1) PRIMARY KEY not null,
sazba_dph numeric(2,2)
)
go
```

```
create table stat
(
id_statu INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (20) not null,
zkratka VARCHAR(3) not null
)
go
```

```
create table jednotky
(
id_jednotky INT identity (1, 1) PRIMARY KEY not null ,
nazev VARCHAR (20) not null,
zkratka VARCHAR (4) not null
)
go
```

```
create table obrazky
(
id_obrazku INT identity (1, 1) PRIMARY KEY not null,
obrazek IMAGE,
alt_nazev VARCHAR (100) not null,
rozliseni VARCHAR (15) not null
)
go
```

```
create table stav_objednavky
(
id_stav_objednavky INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (15) not null,
)
go
```

```
create table stav_platby
(
id_stav_platby INT identity (1, 1) PRIMARY KEY not null ,
```

```
nazev VARCHAR (15) not null,  
)  
go
```

```
create table typ_platby  
(  
id_typ_platby INT identity (1, 1) PRIMARY KEY not null ,  
nazev VARCHAR (60) not null,  
cena numeric(6,2),  
)  
go
```

```
create table zpusob_doruceni  
(  
id_zpusob_doruceni INT identity (1, 1) PRIMARY KEY not null ,  
nazev VARCHAR (60) not null,  
doba_doruceni VARCHAR (60),  
cena numeric(6,2),  
)  
go
```

```
create table barva  
(  
id_barvy INT identity (1, 1) PRIMARY KEY not null ,  
nazev VARCHAR (20) not null,  
)  
go
```

```
create table material  
(  
id_materialu INT identity (1, 1) PRIMARY KEY not null ,  
nazev VARCHAR (30) not null,  
)  
go
```

```
create table velikost  
(  
id_velikosti INT identity (1, 1) PRIMARY KEY not null ,  
hodnota VARCHAR (4),  
)  
go
```

```
create table sirka  
(  
id_sirky INT identity (1, 1) PRIMARY KEY not null ,  
nazev VARCHAR (8),  
)  
go
```

```

create table garance
(
id_garance INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (10) not null,
)
go

```

```

create table vyrobce
(
id_vyrobce INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (30) not null,
)
go

```

```

create table pravomoc
(
id_pravomoc INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (100) not null,
administrator BIT,
zamestnanec BIT,
zamestnanec_omezena_prava BIT,
pravomoc_pridelena date not null,
)
go

```

```

create table zamestnanec
(
id_zamestnanec INT identity (1, 1) PRIMARY KEY not null,
jmeno VARCHAR(25) not null,
prijmeni VARCHAR(35) not null,
email VARCHAR(60) not null,
heslo VARCHAR (50) not null,
datum_registrace date not null,
id_pravomoc INT FOREIGN KEY (id_pravomoc) REFERENCES pravomoc(id_pravomoc),
)
go

```

```

create table kategorie
(
id_kategorie INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (50) not null,
url VARCHAR (100),
popis VARCHAR (500),
)
go

```

```

create table podkategorie
(
id_podkategorie INT identity (1, 1) PRIMARY KEY not null,

```



```

nazev VARCHAR (50) not null,
url VARCHAR (100),
popis VARCHAR (500),
id_kategorie INT FOREIGN KEY(id_kategorie) REFERENCES kategorie (id_kategorie),
)
go

```

```

create table dorucovaci_adresa
(
id_doruc_adresa INT identity (1, 1) PRIMARY KEY not null,
mesto VARCHAR (30) not null,
psc VARCHAR (5) not null,
ulice VARCHAR (50) not null,
c_domu VARCHAR (5) not null,
id_statu INT FOREIGN KEY (id_statu) REFERENCES stat (id_statu),
)
go

```

```

create table fakturacni_adresa
(
id_fakt_adresa INT identity (1, 1) PRIMARY KEY not null,
id_statu INT FOREIGN KEY (id_statu) REFERENCES stat (id_statu),
mesto VARCHAR (30) not null,
psc VARCHAR (6) not null,
ulice VARCHAR (50) not null,
c_domu VARCHAR (5) not null,
)
go

```

```

create table registrace
(
id_registrace INT identity (1,1) PRIMARY KEY not null,
email VARCHAR(60) not null,
heslo VARCHAR(30) not null,
datum_registrace date not null,
)
go

```

```

create table zakaznik
(
id_zakaznika INT identity (1, 1) PRIMARY KEY not null,
jmeno VARCHAR(15) not null,
prijmeni VARCHAR(25) not null,
email VARCHAR(60) not null,
telefon VARCHAR(13),
na_firmu bit not null,
ico VARCHAR(13),
dic VARCHAR(14),
id_doruc_adresa INT FOREIGN KEY (id_doruc_adresa) REFERENCES dorucova-
ci_adresa(id_doruc_adresa) not null,

```

```

id_fakt_adresa INT FOREIGN KEY (id_fakt_adresa) REFERENCES fakturacni_adresa(id_fakt_adresa)
not null,
id_registrace INT FOREIGN KEY (id_registrace) REFERENCES registrace(id_registrace)
)
go

```

```

create table remarketing
(
id_remarketingu INT identity (1, 1) PRIMARY KEY not null,
id_zakaznika INT FOREIGN KEY (id_zakaznika) REFERENCES zakaznik (id_zakaznika),
jeden_az_tri_dny_navstivil_web BIT,
tri_az_sedm_dny_navstivil_web BIT,
vlozili_do_kosiku30 BIT,
newsletter BIT,
)
go

```

```

create table zbozi
(
id_zbozi INT identity (1, 1) PRIMARY KEY not null,
nazev VARCHAR (100) not null,
cena_nakupni INT not null,
cena_prodejni INT not null,
marze VARCHAR(4),
ean VARCHAR(13),
url_adresa VARCHAR (100) not null,
id_vyrobce INT FOREIGN KEY (id_vyrobce) REFERENCES vyrobce (id_vyrobce) not null,
id_garance INT FOREIGN KEY (id_garance) REFERENCES garance (id_garance) not null,
id_obrazku INT FOREIGN KEY (id_obrazku) REFERENCES obrazky (id_obrazku),
id_dph INT FOREIGN KEY (id_dph) REFERENCES dph (id_dph) not null,
id_barvy INT FOREIGN KEY (id_barvy) REFERENCES barva (id_barvy) not null,
id_sirky INT FOREIGN KEY (id_sirky) REFERENCES sirka (id_sirky),
id_velikosti INT FOREIGN KEY (id_velikosti) REFERENCES velikost (id_velikosti),
id_materialu INT FOREIGN KEY (id_materialu) REFERENCES material (id_materialu) not null,
id_jednotky INT FOREIGN KEY (id_jednotky) REFERENCES jednotky (id_jednotky) not null,
kratky_popis VARCHAR (200) not null,
dlouhy_popis VARCHAR (500) not null,
znacka_akce BIT not null,
znacka_sleva BIT not null,
)
go

```

```

create table vratky
(
id_vratky INT identity (1,1) PRIMARY KEY not null,
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi) not null,
)

```

```

create table sklad
(

```

```

id_skladu int primary key not null IDENTITY,
nazev_skladu VARCHAR(30),
)
go

```

```

create table zbozi_na_sklade
(
id_zbozi_na_sklade INT identity (1, 1) PRIMARY KEY not null ,
id_zbozi INT FOREIGN KEY(id_zbozi) REFERENCES zbozi (id_zbozi),
id_skladu INT FOREIGN KEY(id_skladu) REFERENCES sklad (id_skladu),
id_jednotky INT FOREIGN KEY(id_jednotky) REFERENCES jednotky (id_jednotky),
dostupnost VARCHAR (15) not null,
mnozstvi VARCHAR (4) not null,
datum_aktualizace date not null
)
go

```

```

create table objednavka
(
id_objednavky INT identity (1, 1) PRIMARY KEY not null,
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi),
datum_objednavky date not null,
update_objednavky date not null,
id_stav_platby INT FOREIGN KEY(id_stav_platby) REFERENCES stav_platby (id_stav_platby),
id_typ_platby INT FOREIGN KEY(id_typ_platby) REFERENCES typ_platby (id_typ_platby),
id_stav_objednavky INT FOREIGN KEY(id_stav_objednavky) REFERENCES stav_objednavky
(id_stav_objednavky),
id_zpusob_doruceni INT FOREIGN KEY(id_zpusob_doruceni) REFERENCES zpusob_doruceni
(id_zpusob_doruceni),
id_zakaznika INT FOREIGN KEY(id_zakaznika) REFERENCES zakaznik (id_zakaznika),
id_zamestnanec INT FOREIGN KEY(id_zamestnanec) REFERENCES zamestnanec (id_zamestnanec),
)
go

```

```

create table kosik
(
id_kosiku INT identity (1, 1) PRIMARY KEY,
mnozstvi smallint not null,
id_objednavky INT FOREIGN KEY (id_objednavky) REFERENCES objednavka (id_objednavky),
id_typ_platby INT FOREIGN KEY (id_typ_platby) REFERENCES typ_platby (id_typ_platby),
id_zpusob_doruceni INT FOREIGN KEY (id_zpusob_doruceni) REFERENCES zpusob_doruceni
(id_zpusob_doruceni),
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi),
)
go

```

```

create table zbozi_v_kategorii
(
id_zbozi_v_kategorii INT identity (1, 1) PRIMARY KEY not null,
id_kategorie INT FOREIGN KEY (id_kategorie) REFERENCES kategorie (id_kategorie),

```

```

id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi),
)
go

create table zbozi_v_podkategorii
(
id_zbozi_v_podkategorii INT identity (1, 1) PRIMARY KEY not null,
id_podkategorie INT FOREIGN KEY (id_podkategorie) REFERENCES podkategorie (id_podkategorie),
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi),
)
go

create table reklamace
(
id_reklamace INT identity (1, 1) PRIMARY KEY not null,
id_objednavky INT FOREIGN KEY (id_objednavky) REFERENCES objednavka (id_objednavky),
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi),
id_zakaznika INT FOREIGN KEY (id_zakaznika) REFERENCES zakaznik (id_zakaznika),
duvod_reklamace VARCHAR(500),
reklamovano_datum date
)
go

create table pokladni_zarizeni
(
id_pokladny INT identity (1,1) PRIMARY KEY not null,
nazev_pokladny VARCHAR(50)
)
go

create table faktura
(
cislo_faktury BIGINT identity (1, 1) PRIMARY KEY not null,
ico VARCHAR(8),
dic VARCHAR(12),
jmeno_podnikatele VARCHAR(40),
platce_dph BIT,
c_bank_uctu VARCHAR(20),
kod_banky VARCHAR(4),
datum_vystaveni date not null,
datum_splatnosti date not null,
id_typ_platby INT FOREIGN KEY (id_typ_platby) REFERENCES typ_platby (id_typ_platby) not null,
cena_za_zbozi_a_sluzby VARCHAR(10)
)
go

create table EET_uctenka
(
id_uctenky INT identity (1,1) PRIMARY KEY not null,

```

```

oznaceni_provozovny varchar(10) not null,
datum_trzby date not null,
danove_id_podnikatele varchar(10),
bezpecnostni_kod_poplatnika VARCHAR(10),
fiskalni_identifikacni_kod VARCHAR(10),
id_pokladny INT FOREIGN KEY (id_pokladny) REFERENCES pokladni_zarizeni (id_pokladny) not
null,
id_typ_platby INT FOREIGN KEY (id_typ_platby) REFERENCES typ_platby (id_typ_platby) not null,
cena_za_zbozi_a_sluzby INT not null,
)
go

```

```

create table doklad
(
id_dokladu INT identity (1,1) PRIMARY KEY not null,
id_objednavky INT FOREIGN KEY (id_objednavky) REFERENCES objednavka (id_objednavky) not
null,
id_pokladny INT FOREIGN KEY (id_pokladny) REFERENCES pokladni_zarizeni (id_pokladny),
id_zakaznika INT FOREIGN KEY (id_zakaznika) REFERENCES zakaznik (id_zakaznika) not null,
id_zbozi INT FOREIGN KEY (id_zbozi) REFERENCES zbozi (id_zbozi) not null,
datum_cas_transakce date,
udaje_o_trzbe VARCHAR(100),
cislo_faktury BIGINT FOREIGN KEY (cislo_faktury) REFERENCES faktura (cislo_faktury),
id_uctenky INT FOREIGN KEY (id_uctenky) REFERENCES EET_uctenka (id_uctenky),
)
go

```